3 CD EDITION

116-PAGE MAGAZINE & 2 FULL LINUX DISTRIBUTIONS!

From the makers of LINUX FORMAT and LINUXPRO

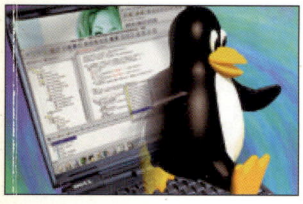
MARCH 2003

Linux

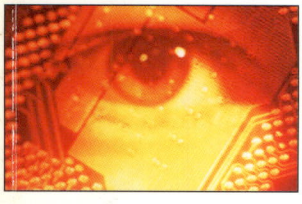
THE COMPLETE GUIDE

BETTER THAN WINDOWS!

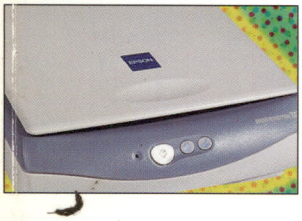
Make your PC more secure and reliable...
Everything you need to master Linux!



PROFESSIONAL GRAPHICS:
WITH OUR GIMP TUTORIAL **PAGE 58**



SECURE YOUR PC:
PROTECT YOUR COMPUTER ON THE INTERNET **PAGE 80**



THE SECRETS OF QUALITY SCANS
HINTS AND TIPS FOR BETTER PICS **PAGE 52**

USING YOUR COVERDISCS **PAGES 10/14**

the future network
Media with passion

9 771476 028010 02
MARCH 2003
£7.99

21 FANTASTIC TUTORIALS

● Install Linux ● Edit graphics ● Burn CDs ● Create presentations ● Compile software
● Manage partitions ● Secure your system ● Use spreadsheets ● And lots more!

Connecting USB digital cameras under Linux! **PAGE 61**

Bravo, Raritan!

Masterful KVM Switches

A full range of server management products

For more than 12 years, the toughest critics – over 25,000 IT professionals – have experienced the beautiful music of Raritan KVM solutions. Every Raritan KVM switch includes a dedicated microprocessor for every port to prevent keyboard/mouse lock-ups and premium circuitry for top quality video. Choose a Raritan KVM switch for ...

Total Scalability – Whether you require access for 1 or 32 simultaneous users, managing 2 computers at your desktop, 30 servers in your data center, or 2,000 systems at a server farm, Raritan has a scalable solution.

Total Out-of-Band Control – BIOS-level access to servers, even when the network is down – no network bandwidth required.

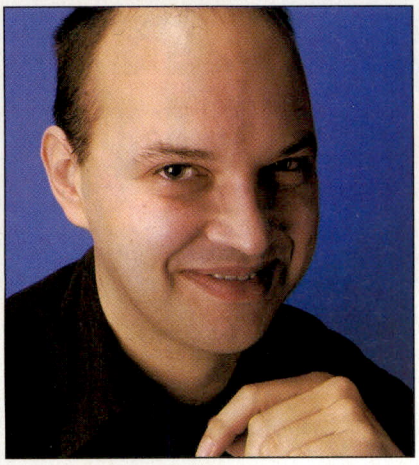
Total Convenience – Plug and play set-up with coax or Category 5 UTP cable, multi-platform support, on-screen user interface, and non-blocked access to all connected devices – without software or a separate administration network.

Total Security – "Bullet-proof" security, with strong 128-bit SSL encryption for KVM access over IP – now, more than ever.

Total Choice of Access Method – For large-scale IT environments, Raritan's Paragon-TeleReach solution, winner of Network Computing's Editor's Choice, provides superior, secure direct access, with KVM access over IP when you need it. Direct, or "over IP"– **the choice is yours.**

Raritan.

www.raritan.com
+31-10-2844040
www.raritan.com



Welcome to the world of free software!

Congratulations on taking the plunge and deciding to find out more about Linux. Although the hype over this superb, free operating system may have died down a bit, that's only because it's no longer news that people are making the switch. Linux is growing in popularity faster than any other OS, and that's partly because people like you are looking for something a bit different.

There are several good reasons why it's a good time to try Linux. The first is that many people are dissatisfied with the alternatives. Other commercial operating systems seem to be moving towards a model where they actually control what you do with your computer. Even more amazingly, they want you to pay regularly for the privilege too.

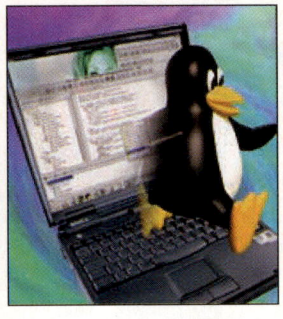
Linux is free, in two senses of the word. You can download versions from the Internet or get it on cheap CDs. It isn't owned by anybody and all the source code is readily available. It's also backed up by a whole load of free software. If you want to manipulate graphics, play MP3s, word process, create presentations, burn CDs or edit video, you can do it all with software that won't cost you a thing.

This may well be the reason why a lot of people investigate Linux, but when they use it, they discover quite a few other benefits. For a start it is very robust and reliable – there are stories of Linux machines running continuously for years without fail. While not completely crash-proof (nothing is), we think you're going to be impressed. It's also, by default, a lot more secure than many other desktop operating systems. You won't be troubled by email viruses for a start, and as various firewall schemes are built into the kernel, you don't need to be so worried about going online either.

However, you'll find all this out for yourself when you get going with the discs provided in this box!

How we can help

Learning how to use a new operating system can be a bit daunting, but Linux is a lot easier to pick up than you might expect. To help you get started you'll find a whole range of tutorials, hints and advice over the next hundred or so pages, and some pointers on where to get further help if you need it. We hope you enjoy your Linux experience!



Nick

Nick Veitch
Editor
nick.veitch@futurenet.co.uk

EDITORIAL

Editor Nick Veitch *nick.veitch@futurenet.co.uk*
Art Editor Julian Jefferson
Production Editor Mark Wheatley

EDITORIAL CONTRIBUTORS

Neil Bothwick, Andy Channelle, David Coulson, Richard Drummond, Marco Fioretti, Maurice Kelly, Biagio Lucini, Neil Lucock, Matt Nailon, Mark Newby, Robert Smith, Charlie Stross

ART CONTRIBUTORS

Chris Winn, Gary Kempston, Getty Images, Superstock

FUTURE DIGITAL

Managing Director Colin Morrison
Publishing Director John Weir

Publisher David Taylor
Senior Editor Dan Hutchinson

EDITORIAL ENQUIRIES

Linux The Complete Guide, Future Publishing Ltd, 30 Monmouth Street, Bath BA1 2BW
Tel 01225 442244
Fax 01225 822744
Web futurenet.co.uk

Media with passion

Printed by Wyndham Heron
All Content © Future Publishing Ltd 2003

Copyright No part of this publication may be reproduced without written permission from our publisher. We assume all letters sent – by email, fax or post – are for publication unless otherwise stated, and reserve the right to edit contributions. All contributions are submitted and accepted on the basis of non-exclusive worldwide licence to publish or license others to do so unless otherwise agreed in advance in writing. *Linux The Complete Guide* recognises all copyrights and trademarks. Where possible, we have acknowledged the copyright holder. Contact us if we haven't credited your copyright and we will always correct any oversight.

All CD-ROM demos and reader submissions are supplied to us on the assumption they can be incorporated into a future covermounted CD-ROM, unless expressly stated to the contrary. We cannot be held responsible for mistakes or misprints. *Linux The Complete Guide* recognises all copyrights in this issue. Where possible we have acknowledged the copyright holder. Please contact us if we have failed to credit copyright.

Disclaimer All tips in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your computer, peripherals or software through the use of any tips or advice.

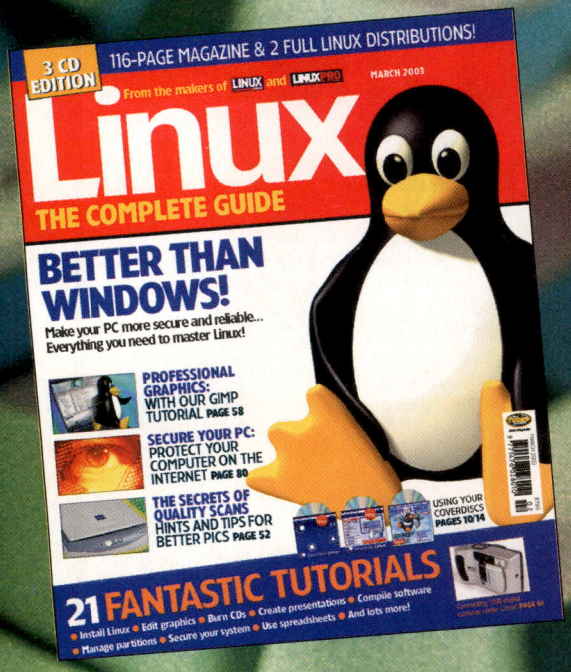
LINUX is a trademark of Linus Torvalds. GNU/Linux is abbreviated to Linux throughout for brevity.

All other trademarks are the the property of their respective owners

Future Publishing Ltd. is part of The Future Network plc.

The Future Network produces carefully targeted specialist magazines and websites for groups of people who share a passion. We aim to satisfy their passion by creating titles that offer superb value for money, trustworthy information, multiple ways to save time and money and are a pleasure to read or visit. Today we publish more than 80 magazines and over 30 magazine websites and networks from offices in five countries. The company also licenses 32 of its titles resulting in over 60 local editions in a further 23 countries.

The Future Network plc is a public company quoted on the London Stock Exchange (symbol: FNET).


3 CD EDITION
116-PAGE MAGAZINE & 2 FULL LINUX DISTRIBUTIONS!
From the makers of LINUX and LINUXPRO
MARCH 2003
Linux
THE COMPLETE GUIDE
BETTER THAN WINDOWS!
Make your PC more secure and reliable... Everything you need to master Linux!
PROFESSIONAL GRAPHICS: WITH OUR GIMP TUTORIAL PAGE 58
SECURE YOUR PC: PROTECT YOUR COMPUTER ON THE INTERNET PAGE 80
THE SECRETS OF QUALITY SCANS HINTS AND TIPS FOR BETTER PICS PAGE 52
USING YOUR COVERDISCS PAGES 10'14
21 FANTASTIC TUTORIALS
Install Linux • Edit graphics • Burn CDs • Create presentations • Compile software
Manage partitions • Secure your system • Use spreadsheets • And lots more!

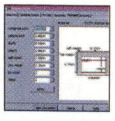
Contents

Everything you need to explore the world of free software! With this magazine and the accompanying CD-ROMs you'll be able to master Linux in next to no time...

6 **WHAT ON EARTH IS LINUX?**
FIND OUT THE ORIGINS AND AIMS OF THIS REVOLUTIONARY SOFTWARE MOVEMENT

10 **MANDRAKE LINUX INSTALLATION**
GET THIS OPERATING SYSTEM RUNNING AND FIND OUT HOW TO GET THE BEST OUT OF IT

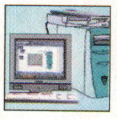
14 **DEMO LINUX GUIDE**
SEE HOW LINUX CAN CHANGE YOUR COMPUTING EXPERIENCE – WITH NO INSTALLATION REQUIRED!



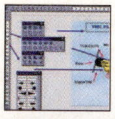
16 **OPENOFFICE.ORG INSTALL**
LEARN ALL ABOUT THIS GREAT SOFTWARE SUITE AND HOW IT RIVALS MICROSOFT'S OFFICE PACKAGE

20 **OPENOFFICE.ORG WORD PROCESSING**
DISCOVER THIS RANGE OF POWERFUL TEXT TOOLS

24 **OPENOFFICE.ORG SPREADSHEETS**
FIND OUT JUST HOW FLEXIBLE THIS EXCEL RIVAL IS

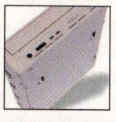


28 **OPENOFFICE.ORG DRAWING**
GET TO GRIPS WITH THE POWERFUL GRAPHICAL OPTIONS THAT OPENOFFICE.ORG PROVIDES



32 **OPENOFFICE.ORG PRESENTATIONS**
LIVEN UP MEETINGS WITH THESE INNOVATIVE TOOLS

37 **THE SHELL**
GET MORE CONTROL OVER YOUR COMPUTER WITH THE INCREDIBLY POWERFUL COMMAND SHELLS



38 **MOUNTING**
GET DEVICES AND DRIVES MOUNTED ON YOUR LINUX MACHINE WITH OUR STEP BY STEP TUTORIAL

40 **PARTITIONS**
SAVE YOUR DATA WITH THIS COMPREHENSIVE GUIDE TO BACKING UP YOUR HARD DRIVE PARTITIONS

44 **FILE PERMISSIONS**
MAKE SURE THAT ONLY THE PEOPLE AND PROCESSES YOU CHOOSE CAN READ YOUR FILES

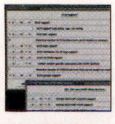
48 **CODE COMPILING**
COMPILING CAN SEEM DAUNTING, BUT OUR BASIC GUIDE WILL HELP YOU AVOID POTENTIAL PITFALLS

52 **SCANNING**
GET YOUR SCANNER WORKING WITH SANE AND IMPROVE YOUR IMAGE QUALITY WITH OUR GUIDE

58 **THE GIMP**
WE INTRODUCE THIS WORLD CLASS GRAPHICS PACKAGE WITH A GUIDE TO COMPOSITE IMAGES

61 **CAMERAS**
CONNECT YOUR USB DIGITAL CAMERAS TO YOUR LINUX MACHINE THE QUICK AND EASY WAY

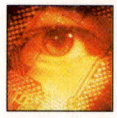
62 **WINEX**
AN IN-DEPTH LOOK AT WINEX – IS IT REALLY THE FUTURE OF LINUX GAMES DEVELOPMENT?



68 **CD WRITING PART1**
BURNING CDS IS FAST, FLEXIBLE AND EASY USING THE COMMAND LINE. WE EXPLAIN HOW YOU DO IT

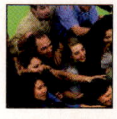
72 **CD WRITING PART2**
BURN MIXED MODE CDS UNDER AN X INTERFACE IN THE SECOND PART OF OUR BURNING TUTORIAL

76 **WIRELESS NETWORKING**
UNTANGLE YOURSELF FROM NETWORK CABLES WITH OUR GUIDE TO WIRELESS ETHERNET



80 **INTERNET SECURITY**
NETWORK CRIME IS ON THE RISE. WE EXPLAIN WHY, AND HOW YOU CAN PROTECT YOUR COMPUTER

86 **WHAT ON EARTH ARE MAN PAGES?**
GET HELP WITH THE HELP FILES!



90 **GET MORE FROM OPEN SOURCE**
FIND OUT HOW YOU CAN PLAY YOUR PART AND CONTRIBUTE TO THE OPEN SOURCE COMMUNITY

96 **ESSENTIAL READING**
ALL OF THE BEST BOOKS TO HELP YOU LEARN MORE ABOUT LINUX, ITS ORIGINS AND CREATORS

102 **ANSWERS**
WE PROVIDE THE SOLUTIONS TO SOME OF THE COMMON LINUX PROBLEMS THAT CROP UP

110 **HELP**
STILL GOT PROBLEMS? OUR HELP PRIMER WILL POINT YOU IN THE DIRECTION OF AN ANSWER

112 **LINUX USER GROUPS**
GET MORE INVOLVED IN THE LINUX COMMUNITY – FIND A USER GROUP IN YOUR AREA TODAY!

What on Earth is... LINUX?

The fact that you're holding a Linux magazine suggests you know a little about it, but how much? Andy Channelle looks at the world of Linux through a Newbie's eyes...

Back in the 1960s, when computers were the size of small houses and could only be operated by men with beards and a stack of hole-punched cards, there was no such thing as an operating system. Programmers would first have to create an environment, then create their application to run in it. It took a lot of card punching, and also meant that an app created on one system would probably not work on another, even from the same manufacturer. This couldn't go on, and in 1969 a pair of engineers from AT&T's Bell Labs created an OS called Unix.

Unix gave users a 'standard' way to interact and control their computers. However, these standards weren't very standard at all; a number of proprietary Unix flavours appeared, created by hardware manufacturers desperate to 'lock in' users to their kit. The only thing these variations had in common was that they were all hideously expensive.

Then, in 1991, a student from Helsinki started to create his own Unix-compliant OS – "just a hobby" the young Linus Torvalds wrote in one newsgroup – that would be released on the Internet and distributed freely as Open Source software.

This project was taken up by a community of geeks, hackers and astute entrepreneurs. Linux, as it became known, grew in scope and sophistication. Although in its initial phase, and only really of interest to other programmers, those programmers began to construct the applications that would ease non-technical users into the Linux world. It's estimated that Linux now resides on about eight per cent of the world's PCs, and is growing faster than any other operating system.

I'm free! Tux does his best John Inman impression.

What does it do?

Like any other operating system, Linux is a program that allows users to control their computers, acting as a host for the applications that make owning a computer worthwhile. From comprehensive office suites and graphics tools to web design and games, Linux is blessed with every type of application you could imagine – and some you couldn't. It is especially popular for 'mission critical' tasks that require stability and security, such as running World Wide Web sites and e-commerce. The downside of such power has been that, in the past, the OS has been regarded as difficult to install and use for the average user. That is changing, however, as the face of Linux becomes more friendly. It is now making real headway in the mainstream.

As well as being a multitasking OS (it can do more than one thing at once), Linux is also multiuser. In an office, this means that one server can handle a variety of applications at the same time: the finance director could browse the company accounts while the secretary secretly played *Quake* in a different room. For home users, multiuser means that each member of the family can have their own set-up – including applications, desktop themes and so on – which won't interfere with everyone else's.

Great! How much does it cost?

The revolutionary thing about Linux, apart from the fact that it doesn't crash every five minutes, is the way in which it is distributed. While the big software corporations charge an arm and a leg to let you use their systems, Linux is free! You can download it from the Internet (if you've got a lot of patience and an unmetered connection), or, you can copy it off your mate or install it from a magazine coverdisc.

Though the profit-motive is a very large aspect of the phenomenal growth of Linux, some credit must also be given to the growing number of companies who have set about creating various distributions which give users not just a top quality OS, but also a range of value-added software, utilities and programming tools too.

The big names in Linux distribution include Red Hat, SuSE, Corel and Debian, but these are just the tip of a very large iceberg. Go to any search engine, type in 'Linux distributions' and prepare to be amazed. Distros vary in both quantity and quality, but most come with a wide range of free or Open Source software and can be bent and shaped to your own particular tastes and needs.

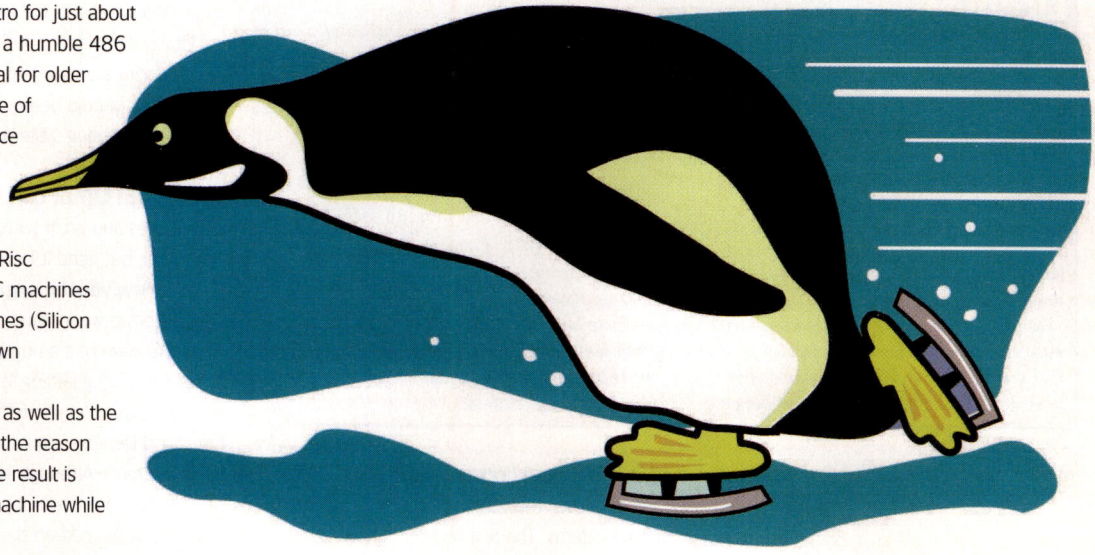
What does it run on?

One of the most useful things about Linux is that it's not tied to a single platform. While Windows will only run on Intel

machines (or clones thereof) and MacOS is only at home on Apple computers, you'll find a Linux distro for just about anything. The x86 version will install on a humble 486 (or even on a 386), which makes it ideal for older machines, but it is also perfectly capable of squeezing every last drop of performance from the latest Pentium III. There are also flavours available for Motorola 680x0 processors (Amiga and Atari ST), Sun Sparc workstations, Advanced Risc Machine processors (Acorns), Power PC machines (iMacs etc), MIPS R3000/4000 machines (Silicon Graphics workstations) and also full blown mainframes like the IBM S/390.

The openness of the Linux kernel – as well as the dedication of hardware enthusiasts – is the reason behind this platform promiscuity, but the result is that you can extend the life of an old machine while still using up-to-date software.

Linux is also quickly becoming the world's system of choice for embedded applications such as DVD players, personal information managers, media terminals and Internet access devices.



Sleek, smart and fast... Just some of the qualities shared by penguins and Linux.

So it's a software program then?

Although it's often referred to as a single program, the average Linux installation is in fact a collection of software, the heart of which is the kernel. In order to make any sense of the kernel, you'd probably need a computer science degree or a really tenacious will to learn. Fortunately, you can simplify things considerably by installing a shell to act as a go-between for the user and the kernel.

There are two types of shell: the command line interpreter (CLI) and the graphical user interface (GUI), and to further confuse things, there are various versions of both types. The CLI is equivalent to MS-DOS in that you issue commands to the OS by typing arcane phrases such as 'echo $SHELL' before sitting back and waiting for the results. The CLI may be a powerful tool and can sometimes be essential for power users, but if you find programming as interesting as concrete, you'll really need a GUI.

In the bad old days of proprietary UNIX, a GUI was only possible with the addition of a costly X Windowing system (the next layer of our Linux cake). However (and you'll notice a common thread here) a community of coders got together and decided to create a free one. This is called XFree86. It handles not only graphics cards and monitor resolution, but also all your mouse actions and keyboard presses.

The latest version, XFree86-4, includes support for the latest generation of 3D graphics cards and chipsets, and makes installing Linux as easy as pie. Well, almost as easy as pie...

What's a Desktop Environment?

Your Desktop Environment is the thing you actually look at all day long. KDE was the original king of the desktops, but disagreements over the direction of the project and, more specifically, the licensing of its source code, led one group of coders to branch out. They created The GNOME Foundation, dedicated to producing free software under the GNU protocol.

GNOME and KDE both do admirable jobs of providing a stable, customisable and intuitive environment for Linux, and also provide an easy means of migration for technophobe Windows users. GNOME, like its rival, is actually more than just a desktop. It provides a development platform, and the tools, for coders to create a range of standards-compliant applications that are then released under the GNU licence. Part of the package is GNOME Office, a set of productivity tools including a word processor (*AbiWord*), a spreadsheet (*Gnumeric*) and a graphics package (*The GIMP*). Taking its cue from Windows, the GNOME applications have a consistent look and feel and, in the case of *The GIMP* in particular, are rapidly becoming just as accomplished as their proprietary cousins.

Both KDE and GNOME are flexible, comprehensive alternatives to Windows, and the great thing is that, if you can't decide which desktop you prefer, you can install both and choose between them to suit your mood – it won't cost you a bean. And if you like the look and feel of Windows, MacOS or even BeOS (to name but three), just point your browser at **www.themes.org**, where you'll find thousands of different 'skins' for both environments.

So what's the bad news?

There is always some bad news. In the Linux world, this comes in the form of the ubiquity of Windows and the desire of PC manufacturers to save money. As PC processors became more

What's the point?

So, if Linux is an operating system similar to Windows or MacOS, what's the point in using it in preference to those other two? The biggest reasons that many people will give involve cost and customisation.

Linux, as we've seen, is distributed under an Open Source agreement, meaning that you can effectively get it for free from the Internet or on any number of cheap CDs. The Open Source licence also gives users or programmers the opportunity to alter or customise any part of the operating system as they see fit. These adjustments are usually then fed back into the community so that other Linux users can make use of them.

Using Open Source software frees users from the constant cycle of costly upgrades that companies such as Microsoft and Apple rely on to swell their coffers. This cost can be a big irritation for home users, but if you're a corporation with thousands of PCs and you discover that your system has just become obsolete, the cost of replacement can run into millions of pounds.

And that brings us on to possibly the best thing about the Linux community: it is a community. Thousands of developers and users are dedicated solely to the cause. They are not happy just to use what works, but instead they are all willing and able to make it better, faster, easier and more accessible.

If you're having trouble with your system, you know you can always call on the experience and knowledge of hundreds and thousands of fellow users who will always be happy to help out with any problems you have.

Licences

The computer world loves its acronyms and, especially when it comes to licences, you could be forgiven for thinking that it's all gone a little bit too far...

GPL (General Public Licence) guarantees your right to copy and change free software. GPL software should always include the source code to allow users to make changes. LGPL is the lesser version of the above which enables linking to non-free software modules.

NPL is the Netscape Public Licence, which is what *Mozilla* is being developed under. It means that though *Mozilla* is free to use and alter, Netscape will be allowed to use any work undertaken on it in their proprietary programmes. Of course, this doesn't work the other way around...

MPL is the Mozilla Public Licence and, like NPL, is a free software licence that has a number of restrictions on it.

X11 is the licence that XFree86 uses.

The Sun Industry Standards Source Licence (SISSL) is being used to develop *OpenOffice.org* in conjunction with the Lesser GPL license.

There are almost as many different licences as their are Linux distributions. If you need or want more information, point your browser at the Open Source website at: www.open source.org.

« powerful, chip manufacturers have decided that the CPU could work a lot harder and take over the running of sound systems, graphics and, crucially, modems. The first two items have been easily integrated into Linux, but the third has proved problematic. These 'winmodems' rely on software drivers that are an integral part of Windows and are unlikely to work with Linux. This is beginning to change – check out **www.linmodems.org** for more information – but it's a slow process, so don't hold your breath if you have one sitting in your machine.

Scanners from certain manufacturers are also impossible to run with Linux owing to said manufacturers refusing to release the information for their devices that would allow others to create the drivers. So if you have something like a UMAX parallel port device, your pictures will have to remain unscanned. Or will they?

Can't I have Linux and Windows?

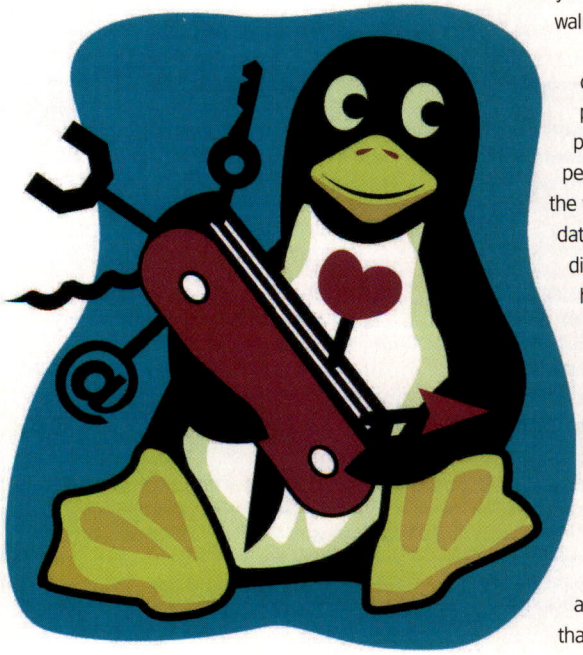
For various reasons (such as the ones given above), many people are reluctant to trash their other OSs. The good news is that you don't have to: Linux will happily run alongside (although not at the same time) as Windows or MacOS.

By giving it a home on your hard drive, you'll be able to scan or surf in your old OS, while getting the benefits of Linux. Linux can get at files from Windows, so you can even use your scans or downloaded wallpaper with *The GIMP*.

The process of installing a dual boot system begins with preparing your hard disk for partitioning. This is where people often have problems as the wrong choice could lose data, but many of the modern distros have special tools to help you, so it's not that daunting. Next, the software formats your newly created partition (Linux has its own file allocation system), and begins adding the operating system and – if you've selected them – a variety of development tools, applications and games. And that really is about it. The next time you boot, you'll be presented with a menu from where you can select which OS to boot up.

Obviously, with PCs being such sophisticated, complicated beasts, there are many problems that can arise, but in most cases the process should be no more taxing than that of installing the Windows operating system.

The right tool for the job. Linux is the Swiss Army Knife of the computer world.



Can Linux run Windows software?

Yes and no. If you'd like to 'go native' and dump Windows altogether, but need it for one or two vital applications, check out *WINE* (**www.winehq.org**), a small program that acts as an interpreter between Windows applications and Linux.

Users have had a number of notable successes with the system, including getting *Word 2000* and *Excel 2000* running. *WINE* does a good job with most things and companies such as Corel and Deneba use it to rapidly port their Windows apps to the Linux platform. Although *WINE* isn't perfect yet – and may never be – it does provide a way for users to keep a vital application without having to keep the OS.

Another alternative is *VMWare*, which is a straight PC hardware emulator that works under Linux. However, *VMWare* is commercial software and so needs to be paid for, and you'll also need a valid copy of Windows.

What's next?

The computer industry moves fast and Linux is no exception. The difference is that the evolution of the OS is both user-led and open to all. Linux development continues at a rapid pace. Development on hardware support has been bolstered by many manufacturers being keen to support Linux now, and the software that runs on Linux is getting better almost daily. Take a look at all the stuff on our coverdiscs for some of the more notable examples.

It's an exciting time to get involved in the community, so what are you waiting for? ■

Software essentials

These are a few of the things that should be on everyone's hard drive. The *GNU Image Manipulation Program*, or *GIMP*, is the Linux world's *Photoshop*, and is an ideal tool for photo retouching, compositing and image creation. As well as the usual suite of brush, pencil and cloning tools, *GIMP* also supports alpha channels and layers, making montages a breeze. *GIMP* can also use plug-ins to increase its functionality. There are over 100 plug-ins freely available on the Internet.

OpenOffice.org Is the Open Source version of Sun's *StarOffice* and has a very accomplished word processor and spreadsheet. Plans were announced recently to integrate *OpenOffice.org* into the GNOME environment. KDE users might also like to try *KOffice* as a fully compliant alternative.

If you've tried *Netscape* and found it to be unreliable, then take a look at *Mozilla* instead. This fast, customisable browser forms the basis of Netscape's 6.1 release and is freely available to download from the Internet at: www.mozilla.org.

Xmms will let you play all those lovely MP3s you've downloaded from the Internet, while *XRipper* will allow you to easily MP3-ify your CD collection.

Civilization addicts can get their fix from *FreeCiv* which, as you'll have gathered, is completely free.

Finally, a number of top quality commercial games – including *Quake III*, *Arena*, *Rune*, *Tribes2* and *Sim City 3000* – will tax the skills of even the most dedicated game player.

Why compromise?
Now you can afford the whole picture.

hp designjet 100

Keep your head and go for the big picture – with the new, versatile HP Designjet 100 colour printer. Because, for the same price as an A2 printer you can now afford a machine that will print sizes up to A1+ – everything from general office documents to CAD drawings – all from your desktop. All with a crisp, sharp, excellent image quality, as only HP knows how. It's affordable, easy to operate and economical to run, using individually replaceable supplies, and operating on special printing modes to control ink usage. You'll also save on trips to the local print shop! And because it's HP, superb reliability and performance come as standard.

• versatile A5 to A1+ output • office printing features: Microsoft® certification for Windows® 98, 2000, XP drivers • dedicated CAD printing capabilities: AutoCAD™ drivers available • easy to operate • saves on outsourcing • supports wide range of hp media • neat desktop design

From **£935** ex vat **£1099** inc vat

To find out more call 0870 443 6281 or visit www.hp.com/uk/giraffe

invent

©Hewlett-Packard Company 2002. Microsoft® and Windows® are US registered trademarks of Microsoft Corporation. AutoCAD™ is a US trademark of Autodesk, Inc. Prices refer to specification descriptions only. Prices quoted are typical buying prices. Prices are correct at time of going to press. Prices change frequently and are subject to change without notice. Additional costs may be incurred for delivery and for credit card transactions. Finance subject to status.

Mandrake 9.0

Neil Bothwick is your guide to the wonders of this month's jam-packed *Linux The Complete Guide* coverdiscs. Featuring the splendid Mandrake 9.0, we've got a full guide to installing and getting the most out of this great Linux operating system.

On the CD

Wherever you see this logo it means there's related stuff on the CD

Important notice

Before you even put a CD in your drive, please make sure you read, understand and agree to the following:

The *Linux The Complete Guide* coverdiscs are thoroughly tested for all known viruses, and is independently certified virus-free before duplication. We recommend that you always run a reliable and up-to-date virus-checker on ANY new software. While every care is taken in the selection, testing and installation of coverdisc software, Future Publishing can accept no responsibility for disruption and/or loss to your data or your computer system which may occur while using this disc, the programs or the data on it. You are strongly advised to have up-to-date, verified backups of all important files. Please read individual licences for usage terms.

READ ME FIRST

Have we got a treat for you! This special issue of *Linux The Complete Guide* features the best parts of the Mandrake 9.0 operating system, providing you with a whole new way of using your PC. Along with a step-by-step guide to installing Mandrake, whether on an individual PC or over a network, we also take a look at some of the fantastic features the OS offers.

With advice on customising your new-look desktop to finding and running the programs you use most often, from playing MP3s to the hugely versatile and handy *Konqueror*, we show you how to personalise your Mandrake installation. We also explain all about the Network Neighbourhood and how to manage your software with easy downloads and installs so your system is always as up to date as it's possible to be.

With all of the great features of this OS here for free, what are you waiting for? Read on and install Mandrake on your PC right now!

Installing Mdk 9.0

You have two alternative ways of installing Mandrake 9.0. The easiest is direct from the CD, which is what we will cover here. The other is a network install by sharing the contents of the disks across a network connection from another PC. Getting started with this option is covered on these pages, but the main install process is the same whichever you choose. CD users simply boot from the first disc.

On booting from the disc, you will see the initial splash screen where you can press **Enter** to proceed or **F1** to set options. For example, you can change the resolution of the graphical install from its default of 800x600 to 640x480 or 1024x768 by typing **vgalo** or **vgahi** respectively. This is useful if you have an LCD monitor as they don't cope well with resolutions other than their default. You can disable the graphical install altogether with **text**.

You have a choice of a clean install or of upgrading an existing Mandrake system. At the same time you need to choose between the "Recommended" or "Expert" install modes. Expert gives you more choice, while Recommended makes many of the decisions for you. If you are new to Linux and have already created some empty space on your hard disk, you may wish to take the Recommended option. Most users with some Linux experience will probably prefer Expert, even if they don't consider themselves experts. Most of the choices have the same defaults as you would get with a Recommended install, but you do get the opportunity to make changes if you wish. We will walk through the Expert install here.

Press F1 at the initial splash screen for extra installation options.

After selecting mouse and keyboard, you need to choose the security level. It is probably easiest to leave it at Standard here, although this isn't really high enough if you are connected to the Internet, so change it later in the *Mandrake Control Centre*.

The next stage is to create the partitions on your hard disk. In Recommended mode this is taken care of automatically. To do the same in Expert, click the "Auto allocate" button in the *DiskDrake* window and select the "simple" option. This will create suitably sized root, home and swap partitions using the empty space on your drive. If you are comfortable setting up your own partitions,

Installing over a network

Harnessing transfer protocols

Mandrake doesn't allow you to export CDROMs via NFS, but it is still possible to install over a network, providing the computer is running a web or FTP server. We'll take a quick look at how you do this with *Apache*. You'll need to be root to do this. First of all, open /etc/httpd/conf/apache.conf in your favourite editor and add this line

```
include conf/mdk90.inc
```

Then type these lines into a new file and save it as /etc/httpd/conf/mdk90.inc

```
Alias mdk90/ /mnt/cdrom/
<Directory /mnt/cdrom/Mandrake>
        AllowOverride All
        order allow,deny
        allow from all
</Directory>
```

and restart *Apache* with

```
apachectl restart
```

You could have added the lines directly to *Apache's* configuration file, but this way is cleaner and means you can easily remove it by commenting out one line.

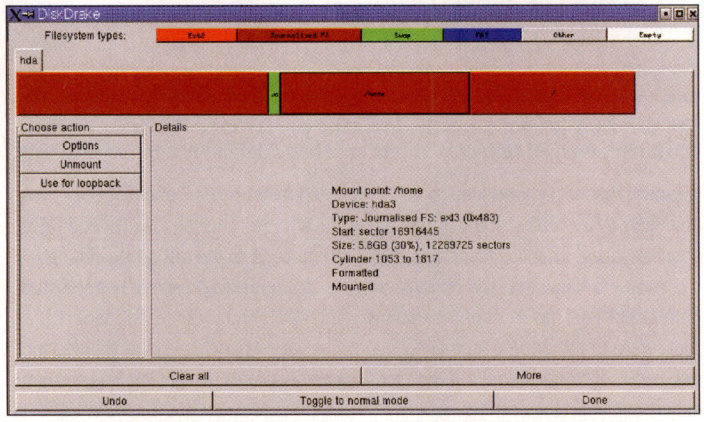
Now you need to create a boot disk using either network.img or pcmcia.img from the images directory of the CD. The former file also includes drivers for some PCMCIA network cards, so it's a matter of trying each to see which works with your setup (my NetGear FA411 card needs pcmcia.img). Create the disk in the usual way:

```
dd if=/mnt/cdrom/images/network.img
of=/dev/fd0
```

Now boot from the disk, making sure the CD is mounted on the server computer. After giving the network information (hostname, IP address, DNS and gateway addresses) for the target machine, you'll be asked for the server details. Give the machine's IP address and use "/mdk90/" for the path.

The install will now proceed just as if you had booted from the CD. The only difference is that you have already given network information, so you won't need to set this up later.

DiskDrake allows you to do this too. Click on "Done" to save the partitions and proceed to formatting. If you have no empty space, *DiskDrake* is able to resize Windows FAT32 (but not NTFS) partitions. See the tutorial on the first disc for more information.



Disk partitioning – Windows partitions can be resized to make room for Mandrake, but read the instructions in the tutorial directory first.

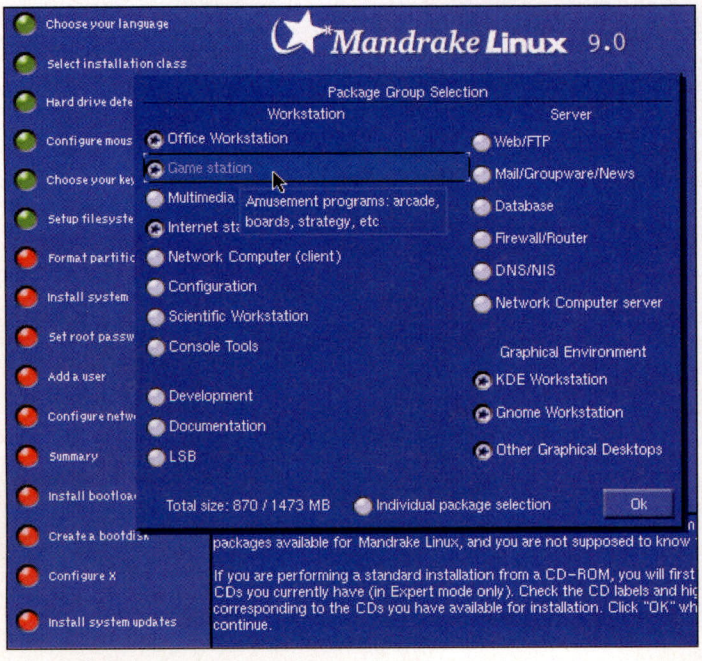
Selecting packages

Now it's time to select the packages you wish to install. Tick the groups of packages you need as there may be some that aren't obviously necessary. You should select Development, even if you are not a developer, so you have all the tools needed to compile packages from source. If you have a network, select the Network Server group if you want to be able to share files or printers with other computers. There's no need to select everything "just in case" as it is easy to install anything you need later from the *Software Manager*. Now go and put your feet up while the packages are installed, although you'll have to pop back to change discs when asked. Package installation can take a while, plus there is some basic config to do.

The install creates an account for the root user and prompts for a password, then you can create additional users. It is important to create at least one normal user as you shouldn't log in as root except when you need to.

You should normally accept the offer to set up a bootloader – this is what starts up Mandrake Linux when you boot. If you have Windows installed, the installer will create a menu to choose between it and Linux at boot time. If you decide to skip this step, you must create a boot disk at the next stage or you won't be able to use your Linux setup. A boot disk is a good idea, even if you set up the boot loader. If you reinstall Windows, it will overwrite the bootloader and you'll need a way to get it back. A boot disk is one method. The other is to boot from the install CD, press **F1** and type **rescue**.

If you set up networking or your Internet connection, you now have the opportunity to check for updates. If possible, you should do this now. It not only makes sure your system is secure, but it also sets up the *Software Manager* for future updates. »



Choose packages in groups, based on your intended use of the computer...

Mandrake Linux 9.0
Choose your language
Choose the packages you want to install
Apache-ASP
Bastille
Bastille-Chooser
Bastille-Curses-module
Bastille-Tk-module
BitchX
C++2LaTeX
CJK
CJK-emacs
Chinput
DansGuardian
Epplets
Eterm
Eterm-devel
FaxMail
Info
Name: BitchX
Version: 1.0-0.c19.3mdk
Size: 2158 KB
Importance: very nice
This is the bleeding edge of IRC software -- the most common functions normally done by scripts are coded into the client itself. It contains dozens of features such as:
* Built-in ANSI color (this is probably the biggest feature)
* Ease of use -- dozens of useful command aliases to reduce typing
* Built-in notify, protection, and bot lists
* Built-in mass commands and tools
* Extended set of DCC commands
Show automatically selected packages
Total size: 7 / 418 MB
Install
Cancel
Create a bootdisk
Configure X
Install system updates
Finally, depending on whether or not you selected individual pac
presented a tree containing all packages classified by groups and
browsing the tree, you can select entire groups, subgroups, or i
Whenever you select a package on the tree, a description appear
selection is finished, click the "Install" button which will then laur
process. Depending on the speed of your hardware and the num
to be installed, it may take a while to complete the process. An e

...and you can fine tune that choice by adding or removing individual apps.

« And that's it, you're there. Remove the CD from the drive, insert your boot floppy if you didn't install a bootloader and let the computer reboot.

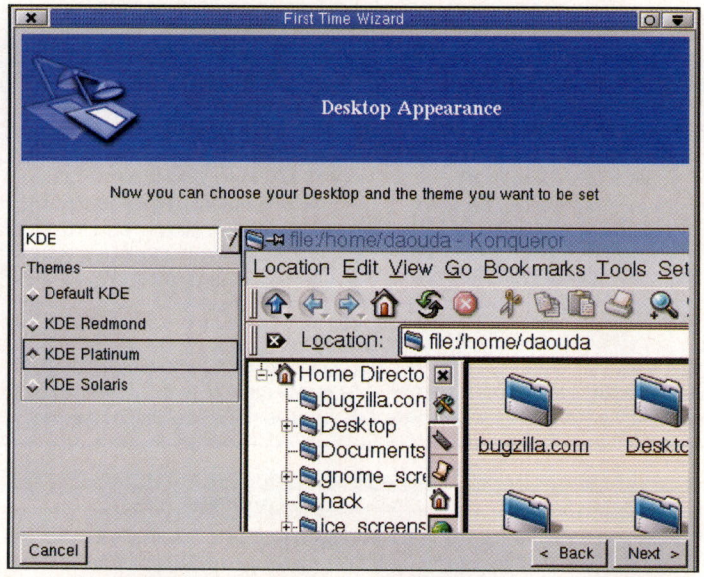
We'll now take a look at some of the features of your new OS.

What you installed

Now that you have installed Mandrake 9.0, let's take a look around. The first time you start Mandrake, the *First Time Wizard* appears. This creates some basic settings and then collects info to register you with Mandrake's website. If you don't want it to send your personal information, or you have already done this on a previous install, you can press the cancel button at this point.

Once the KDE desktop has loaded, click on the KDE menu button at the bottom left of the screen. One glance should be enough to tell you that we won't be able to cover this in a page or two, so we will just pick out some highlights. You can find further information in the tutorial directory of the first CD.

One of the first places you should know about is the *KDE Control Centre*.



The *First Time Wizard* is run the first time you log in to Mandrake (in case you hadn't worked that out). It sets up a basic default config for each user.

It is probably not a good idea to go changing every setting you can find as soon as you've installed Mandrake, but it is useful to know that the options are here. LookNFeel has most of the options that relate to the appearance and behaviour of the desktop. It is safe to experiment with the settings here as each panel has a Defaults button to go back to the original settings, and Reset to go back to the last saved setting.

It's not always necessary to start up the *KDE Control Centre* to make a change – many aspects of the desktop can be altered by right-clicking on them. Right-click on the taskbar to change its settings, size or position. Do the same on the desktop to change its appearance. You can also drop picture files onto the desktop to instantly change the wallpaper.

For some reason, Mandrake has omitted the logout button from the panel in this release. While it's easy enough to logout from the KDE menu or by right-clicking the desktop, you can also put it back where it was in previous releases. Right-click on the panel and select Add>Applet>Lock/Logout applet from the popup menu. If you don't like where KDE has placed it, right-click on the bar to the left of the button and select move, slide it to where you want and left-click.

Tasks, not programs

The programs accessible from the KDE menu are grouped according to function, such as Multimedia, Office, Configuration and so on. Under these groupings you will find the programs listed by name, which is fine if you know the name of the program that does the job you want done. The "What to do?" menu item contains entries for common tasks. For

Getting more out of Mandrake

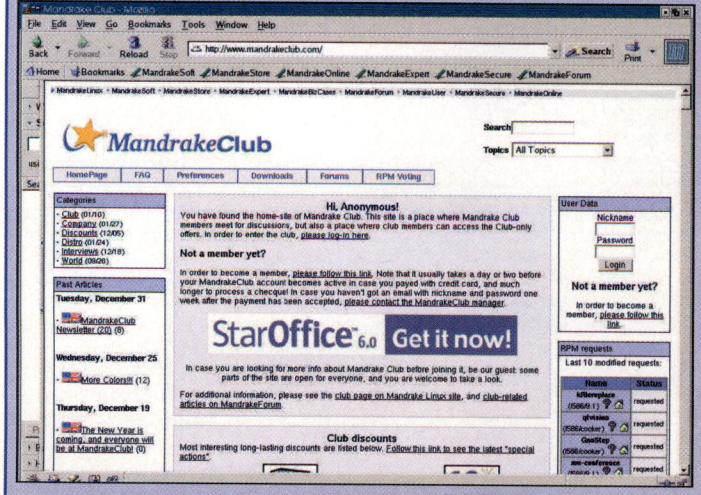
Your guide to the active Mandrake community

One of the great things about Mandrake being so popular is that there is a very active Mandrake user community. These users range from experts to Linux 'newbies' but they all make their impact on the community as a whole. Mandrake themselves have devoted a lot of energy to providing a host of sites for users to express their views, solve their problems and just exchange ideas.

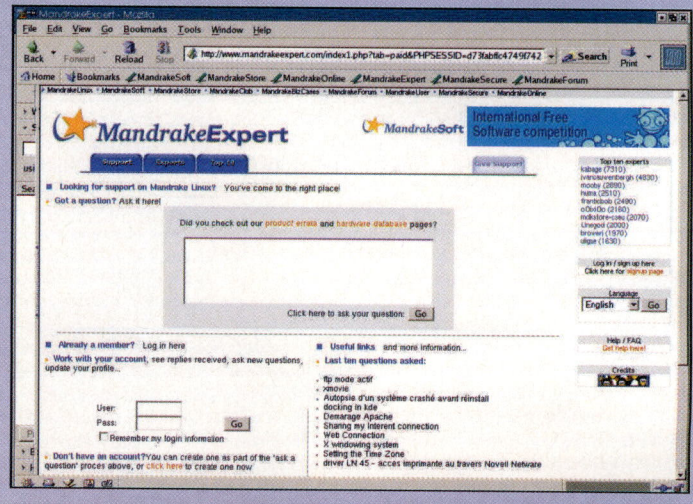
One of the first sites you should visit is www.mandrakeuser.org. This hosts a fairly sizeable collection of guides to various aspects of configuring and using Mandrake Linux, ranging from newbie tasks to advanced sysadmin topics. Get to grips with the *CUPS* printer system, learn more about mounting devices and troubleshoot your networking problems.

If you have a specific problem you can't find an answer for, you should try the Mandrake Expert site, found at www.mandrakeexpert.com. It is designed to process queries efficiently. Ask your question and a whole range of experts will compete to be the first with an answer. Most questions are answered the day they are asked, and with a huge range of experts to answer the questions, even the more obscure problems are usually sorted out.

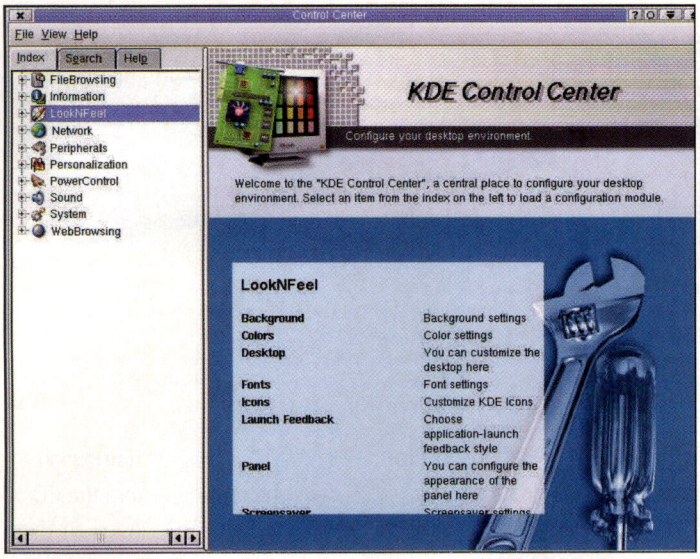
Finally, you should definitely check out the Mandrake Club site, which is found at www.mandrakeclub.com. This contains lots of news and information about the goings on at Mandrake, and special offers and links for club members. You can sign up to the club online for as little as $60 per year and you'll then be able to take advantage of discounts on selected commercial software, vote for your favourite applications to be added to Mandrake and take part in the lively forums.



Sign up to the club for special offers, voting rights and more!


MandrakeExpert
MandrakeSoft
International Free Software competition
Looking for support on Mandrake Linux? You've come to the right place!
Got a question? Ask it here!
Did you check out our product errata and hardware database pages?
Click here to ask your question: Go
Already a member? Log in here
Useful links and more information...
Last ten questions asked:
Work with your account, see replies received, ask new questions, update your profile...
Don't have an account? You can create one as part of the 'ask a question' proces above, or click here to create one now!

All your Mandrake problems can usually be solved in double-quick time.



KDE *Control Centre* lets you change just about any aspect of KDE's operation.

example, to play MP3s, select "Enjoy music & video>Listen to music files". This starts the *XMMS* music player, just as if you had run it from the Multimedia >Sound menu, but it didn't require you to know that the MP3 player is called *XMMS* – would you have guessed that was the name if you didn't know already? The *What to do?* menu should keep you occupied for some time, exploring the various options and programs which are available.

One of the KDE desktop's most useful features is hidden away in the docs. Press **Alt-F2** and you'll get a small window where you can type a command, file, path or URL. *Konqueror* will open a window displaying this, whether it is a web page, FTP site or local directory. It maintains a history of locations typed here, so you only need to type the first few characters the next time. If you type **#** followed by the name of a command, such as **#rpm**, you will be shown the man page for that command, formatted as a web page.

Konqueror is able to handle several other types of "file". If you don't have a column of icons down the left hand side of the *Konqueror* window, press **F9** to add it, then click on the Services icon which is second from the bottom.

Put an audio CD in the drive and select Audio CD Browser. The tracks on the CD are listed in the main window as though they were audio files. Open the Ogg Vorbis folder and you'll see the tracks listed as if they were Ogg Vorbis files. Copy one elsewhere and it will be encoded as it's being copied.

Neighbours

A question often asked when moving from Windows to Linux is, "Where is the *Network Neighbourhood?*" The answer is that it is just below the Audio CD Browser, called LAN Browser. You may need to do some config before you can use it. First, check that you have a package called *lisa* installed. Either use the Software Manager or type **urpmi lisa** in a console. Then start up the *KDE Control Centre*, go to Network>LAN Browsing and click the "Guided LISa Setup" button. After you have gone through the settings, click Apply. Now you can use the LAN browser to view *Samba* shares and FTP servers on your network.

During installation, you were given the option to have one user logged in automatically on startup. This is a convenient feature, although it is less secure if others have physical access to your computer as they can switch it on and be logged in as you. This, and many other items, can be controlled from the *Mandrake Control Centre*.

This is not to be confused with the *KDE Control Centre* – it is a completely different program, launched from the KDE menu or an icon in the taskbar. The *Mandrake Control Centre* has to be run as root, so you'll be asked for the root password. To change the automatic login, click on the System icon and then Users. Here you can make various changes, including enabling and disabling automatic logins. The *Mandrake Control Centre* is also the place to configure networking, hardware or the display resolution.

Software manager

Towards the end of the installation process, you were asked to select a server from which to download updates. In addition to being used at that time, the server was also added to the list of sources used by Mandrake's *Software Manager, rpmdrake*.

The other sources in there are the installation media. This makes installation of further software or dependencies from Mandrake's sources much easier. For example, let's say you didn't select *MySQL* when installing Mandrake but now need it. There are several dependent packages involved, which would each need to be sourced and installed using the standard rpm command. With Mandrake, you can do it in one step, either from a GUI or the command line. The Software Manager GUI is started from the KDE menu at Configuration>Packaging>Install Software. Type **mysql** in the search gadget to get a list of matching packages. You only need to select the main *MySQL* package – *rpmdrake* will sort out the dependencies. To install from the command line, type:

```
urpmi mysql
```

It gets better, because you can add extra sources to the list. This is done from K>Configuration>Packaging> Software Sources Manager or by using **urpmi.addmedia** from the command line. If you are a member of Mandrake Club you will have access to newly created RPMs before they go in to the public contrib area; the Mandrake Club website tells you how to add this source. The contrib area contains packages that are useful, but not part of the main distribution. You can add this by typing:

```
urpmi.addmedia -h Contrib ftp://
ftp.mirror.ac.uk/sites/sunsite.uio.no/
pub/unix/Linux/Mandrake/Mandrake/
9.0/contrib/RPMS
```

This is for the main UK mirror; you can find your local mirror from the Mandrake download page at **www.linux-mandrake.com/en/ftp.php3**. You will usually need to go up a directory from the link here to find the contrib directory. Now let's add another:

```
urpmi.addmedia plf http://
www.zoreil.com/mirrors/www.plf.org/
9.0/i586 with ../hdlist.cz
```

Once again, you'll need to visit the site to find the best mirror to use, and it gives the *urpmi.addmedia* command for each mirror so you can paste it instead of typing. The Penguin Liberation Front (PLF) contains packages that could not be included in the Mandrake distribution for legal reasons, often relating to US law. This includes packages like *libdvdcss*, needed to read CSS-encrypted DVDs.

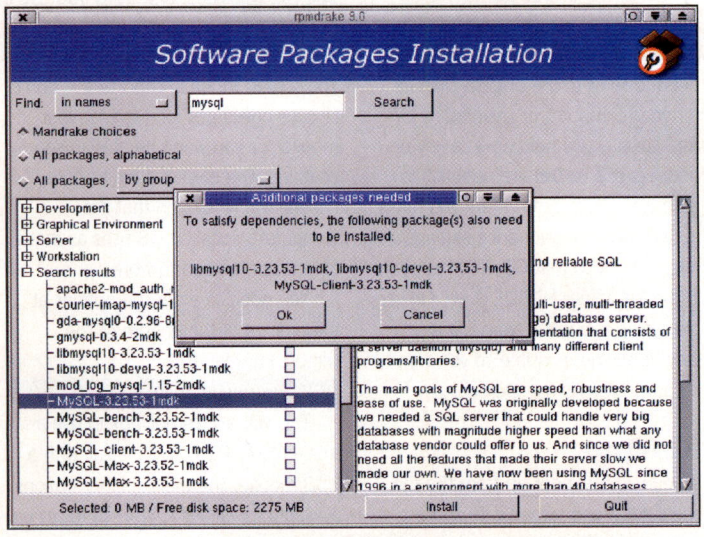
It doesn't matter which site has a package because *urpmi/rpmdrake* takes care of that, so typing

```
urpmi dvdcss
```

will locate and install the packages needed to read CSS-encrypted DVDs. Lists of the available packages are stored locally, so you will need to keep them up to date with

```
urpmi.update -a
```

either from the command line or run from *cron*. ■



The *Software Manager* (*rpmdrake*) takes care of installing all package dependencies, even if they are located on a different server.

DemoLinux

Nick Veitch guides you through using DemoLinux, the Linux distribution you don't need to install!

On the CD

Wherever you see this logo it means there's related stuff on the CD

Important notice

Before you even put the CD in your drive, please make sure you read, understand and agree to the following:
The coverdiscs are thoroughly tested for all known viruses, and is independently certified virus-free before duplication. We recommend that you always run a reliable and up-to-date virus-checker on ANY new software. While every care is taken in the selection, testing and installation of coverdisc software, Future Publishing can accept no responsibility for disruption and/or loss to your data or your computer system which may occur while using this disc, the programs or the data on it. You are strongly advised to have up-to-date, verified backups of all important files. Please read individual licences for usage terms.

READ ME FIRST

One of the things which often prevents people from trying out Linux is their fear of somehow damaging their computer. While it is impossible to cause any physical damage to the computer itself, it is, however remotely, possible to lose some of the data on your hard disks if you start messing around with partitioning.

DemoLinux is an ideal introduction to the world of Linux because there is no need to install it at all. This magical distribution can run completely from the CD without the need to install anything on your computer. Not only will your data be safe, but it means that you can use Linux on virtually any computer.

What you need

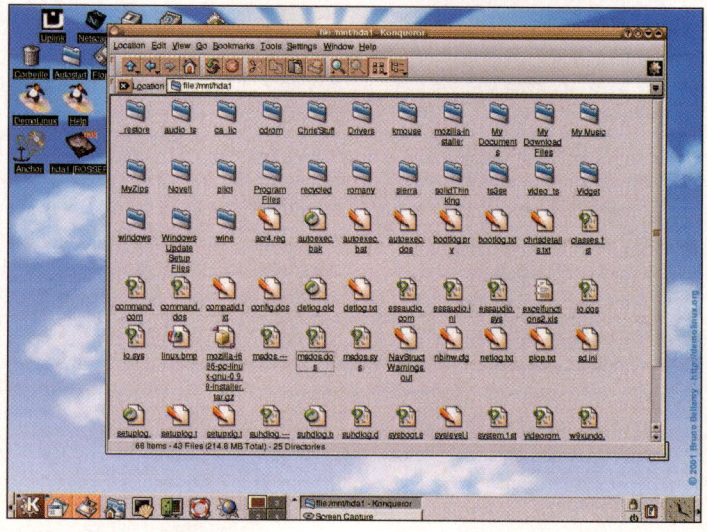
All you need to run DemoLinux is an IBM-compatible PC with a pentium processor, a CD-ROM drive and a reasonable amount of memory. DemoLinux will need 128Mb of memory, although more is better since a lot of the system services and filesystems have to run from RAM.

To boot up from the CD, your PC must be configured to allow booting from a CD drive. The default state of this setting varies so it's something you'll have to check out for yourself. The easiest thing to do is just put the DemoLinux disc in your CD drive and switch on your computer – if it boots up into Linux, off you go!

If not, it's likely that you need to adjust the BIOS settings. You may need to refer to the original manual for your computer or mainboard for this. Usually you hold down a key on boot-up (**F2** or **Del** are popular) to enter the BIOS. From there you need to find the boot-sequence settings and adjust them so that CD-ROM appears in the list before HD/IDE drive. On very old systems, booting from CD is sometimes not possible so you won't be able to easily use DemoLinux.

Startup

When you first boot up from the CD, you'll get a text screen with some instructions. This is so you can set the display mode for the session and activate a USB mouse if you have one. The screen mode is important – don't choose one that's too high for your monitor or you won't be able to see anything. The easiest thing to do is choose the resolution that the computer's installed OS uses as that's almost guaranteed to work.



The file manager will help you use and manipulate all your data.

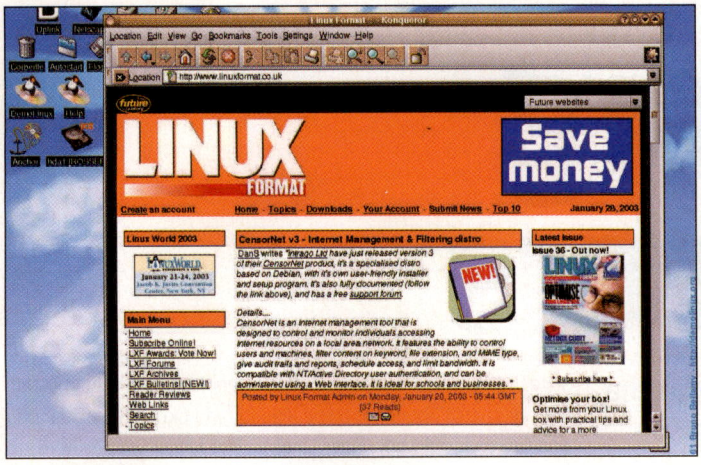
You need to enter the options at the prompt at the bottom of the screen. For example:

```
1024vesa usbmouse
```

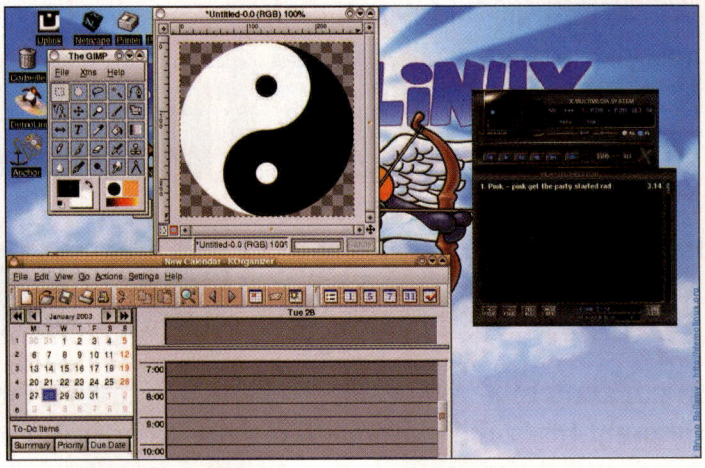
This will tell DemoLinux to use a true-colour 1024x768 display with a USB mouse. There are further options available (press **F3** for these) to set up parameters for other devices. If you are connected to a network that automatically assigns addresses via DHCP (like most broadband services and corporate networks) then you may want to add the 'dhcpclient' option.

When you have entered the options you want to use, press **Return** and the startup will continue.

DemoLinux seems to take a while to start up and that's because it is actually performing a sort of mini-installation too. It has to discover all the hardware connected to your computer and load the appropriate drivers for it. Also, it is running from the CD drive, which is a lot slower than a hard drive. You can amuse yourself by trying to read all the hardware messages as they scroll past!



Konqueror is a fully-featured web browser too.



Try out a whole host of free software included on the CD

Next you will be asked to select your language and keyboard, and you'll be prompted for network information if you have a network card. Make your choices and DemoLinux will continue to load.

You will soon come to the login screen. Linux is quite security conscious by default, so it will always ask for a username and password. DemoLinux is set up with the user 'demo' and a blank password. Enter these and click 'Go!' to start your Linux session. See the boxout on Sessions for more information on the choices.

KDE

If you choose the KDE session, you'll be greeted with a colourful desktop with lots of icons and a launcher bar along the bottom. DemoLinux will also mount all of the hard drives it can find on your system, along with whatever labels you have used for them.

It understands the FAT filesystem used by most versions of MS Windows, and you will be able to read and write files to any of these drives just as you would under Windows. However, be careful – just as in Windows, you will be able to delete files.

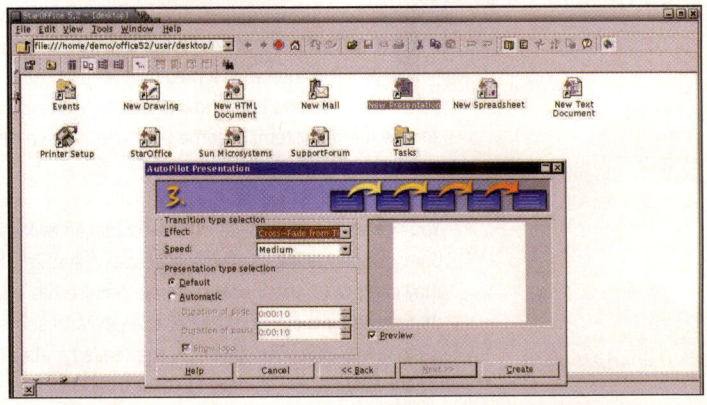
The drive names will be prefixed with the actual device name and partition number (such as hda1). This isn't normally shown in Windows, but it is very useful to know which specific drive you are dealing with.

Clicking on a drive will open it up in *Konqueror*, the file manager. From within *Konqueror*, you can easily view a number of different file types and perform common operations like copying, moving and so on. *Konqueror* is also the default web browser in KDE. Type in a web URL in the "location" bar at the top of the window and you can browse websites too.

GNOME

The other popular desktop environment for Linux is GNOME. This is actually the 1.4 version of GNOME, as the 2.0 release came after this version of DemoLinux was finalised. You can still use most of the popular GNOME tools and get a feel for the environment here though.

Whichever desktop you decide to use, have a go with the small sampling of Linux applications that come with DemoLinux. There's everything from email clients and spreadsheets to calendars, ICQ clients, graphics, MP3/OGG players and plenty of games to choose from.



StarOffice 5.2 is included, for your office requirements.

Take a look at the applications menu in GNOME or KDE by clicking on the icon in the bottom left of the taskbar at the bottom. A menu will appear with all the included software put into convenient categories. Why not fire up the *GIMP* graphics editor and try out the tutorial elsewhere in this magazine?

When you are finished playing with DemoLinux, it is best to use the Logout option from the menu bar. This will take you back to the login screen where you can click on the button to shut down. Although DemoLinux runs from CD and won't be damaged if you simply turn off the computer, any files or discs that you have accessed may be, so it's best to play safe.

Final note

Although DemoLinux is very good for familiarising yourself with the tools and opportunities that Linux offers, it isn't really comparable to the real Linux experience of having an installed version. It will seem a lot slower, especially as you use applications that require a fair amount of memory. Although very clever, a lot of data still has to be read from the CD, which really slows down the launching of applications and their use.

The fact that a lot of data is also stored in RAM means less is available for Linux and so some applications are slowed even further. There are instructions in the DemoLinux guide for permanently installing to hard disk, but if you decide to do this, you're probably better off using the Mandrake discs that come with this magazine. ■

Sessions

Choose your perfect desktop environment

The 'sessions' at the login prompt control the default window manager and desktop environment. Linux allows you to have a variety of these, and you can choose from them every time you log into the system.

These different sessions change the default look and feel of the desktop environment, and often have different shortcuts, themes and menus. Which one you use most often comes down to personal preference. In many ways, the two most popular desktops, KDE and GNOME, are very similar. The Twm session is a special case, designed to make the most of limited resources. If you find the screen refresh too slow or the desktop unresponsive, it may be your hardware just isn't up to the task, in which case Twm will be a better bet.

There is also a special session in DemoLinux that starts *StarOffice 5.2* directly so you can create documents and spreadsheets faster. This is useful if you just want to use Linux to create and print documents, but you won't be able to run any other software easily.

WORKING WITH OPENOFFICE

OpenOffice.org for power users

In our comprehensive guide, Neil Lucock takes a look at the different tools and features that make OpenOffice.org so good, and showing you how to get the most from it.

The procedures described on these pages were done on a 700MHz Pentium 3 running KDE 2 under Mandrake 8.2. This tutorial assumes that you are using an Intel/AMD machine and know how to do the basics, such as navigate around the filesystem, become Root and make a new directory.

What you get

You get tongue-tied, because they chose an awkward name. The phrase "open office" perfectly describes the suite. Unfortunately they can't call it that because someone else has already grabbed that name for their suite. Rather than think of something new, (*Office Libre*, *Buro Frei*, anyone?) they called it after their organisation, so we all have to call it the *OpenOffice.org* suite to avoid legal problems. We call it *OOo* in print to avoid repetitive strain injury from excessive typing.

Although it does not include the *Adabas* database that is included in *SO*, *OOo* has nearly every productivity tool you would need in an office environment. For something that's a free download, it's really the most amazingly good piece of software I've used. It isn't perfect – there are still remnants of *StarOffice 5.2's* interface design that have not yet been replaced, but there is now an effective free office suite that will allow a business to read its *Word* documents and *Excel* spreadsheets. You can still use that CD of clipart too. *OOo* uses windows metafiles (wmf) for its own clipart, it imports *Photoshop* psd files and has import filters for many common file types. However, it's one thing to list a file type under the "import" or "open" dialogues – I'll be looking to see how well the files created by the original programs open in the appropriate tutorials. I was impressed when I wrote the July *Linux Format* review of the presentation software element in the suite. If the *PowerPoint* import filter is representative then most of the arguments about not being able to read old correspondence and use your old work under Linux are no longer valid. A word processor that imports *Word*; a spreadsheet that knows *Excel*; a presentation tool that reads *PowerPoint* files, all contained in the same interface, make very good reasons to look hard at *OOo*.

Single user setup

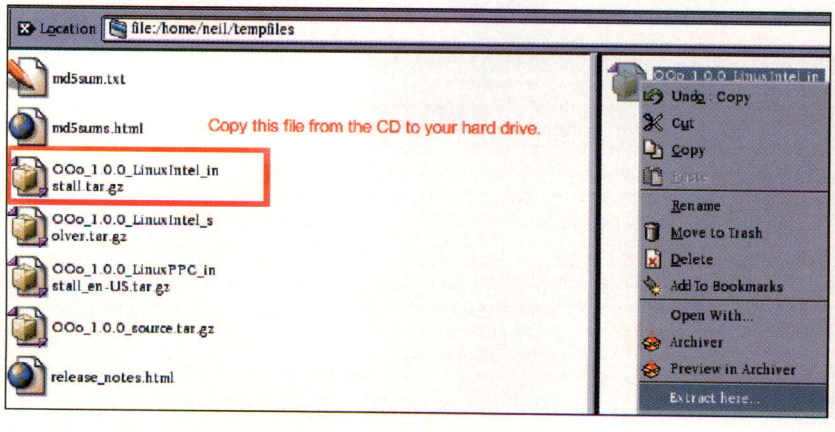
Before we describe this, if more than one person uses your Linux box, or if you want to install to a user account, you will be better off using the "Network" install described later.

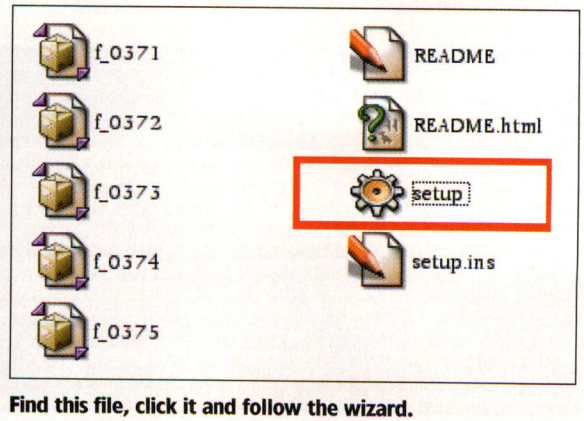
Very few Linux programs seem to have the one-click setup procedures that you find under Windows, but it's really not difficult. The file you need can be download from the Internet at www.openoffice.org (if you have a fast connection or a lot of time) or you can get it on a cheap CD from various distributors. Find the directory and the file 'OOo.1.0.0_LinuxIntel_install.tar.gz'.

Find a place on your hard drive (I have a 'tempfiles' directory for this sort of thing) and copy the file there. It is useful to split the file manager window into two view-panes for this. Once it is on your hard drive, right-click on it and use the 'extract here' option. (If this does not work, you should navigate to where your system keeps the tool you use for making or extracting compressed archives, launch it and dump the contents of the tar.gz file into your temporary directory.) You will then have a lot of files in the directory. Ignore them all except for the file called 'Setup'. Click this and it launches the installation program. From then on, just follow the prompts and click on the 'next' buttons. I did the 'standard installation' which put 183.6MB of files onto my drive. At the end of the process, it tells us that *OOo* is now on our hard drive, but how do we launch it?

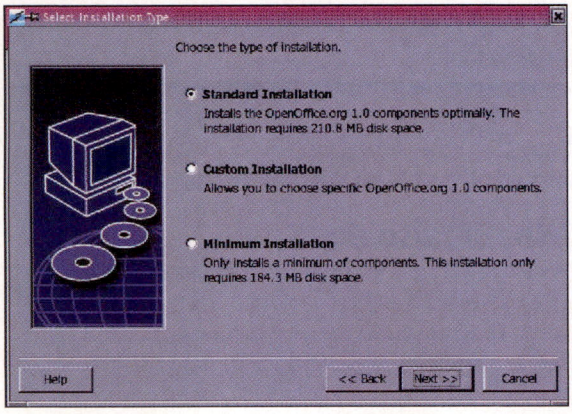
There are two easy ways to put an icon on the desktop. If we right-click, one of the options is to create a new "link to an application". Once we have told it what name we want to use, we tell it where to find the program. *OOo* should have made a

Home user setup. Choosing the right file to install always helps. Extract everything to a temporary directory.

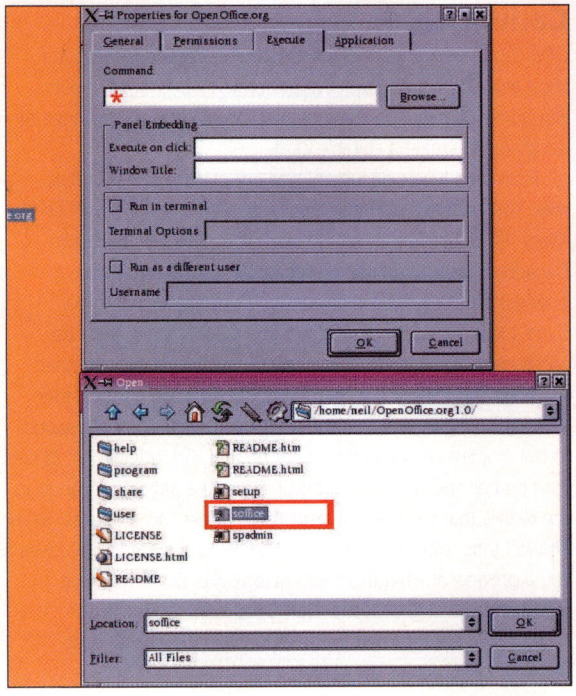




Find this file, click it and follow the wizard.


Select Installation Type
Choose the type of installation.
Standard Installation
Installs the OpenOffice.org 1.0 components optimally. The installation requires 210.8 MB disk space.
Custom Installation
Allows you to choose specific OpenOffice.org 1.0 components.
Minimum Installation
Only installs a minimum of components. This installation only requires 184.3 MB disk space.
Help
<< Back
Next >>
Cancel

The wizard makes installing simple. Choose Custom if you are short of space and don't want some parts of the suite.



Making a link for your desktop. Make sure you type the correct path into the execute box. The path to the file is displayed at the top of the 'Open' dialogue box.

History Lesson

A star is born

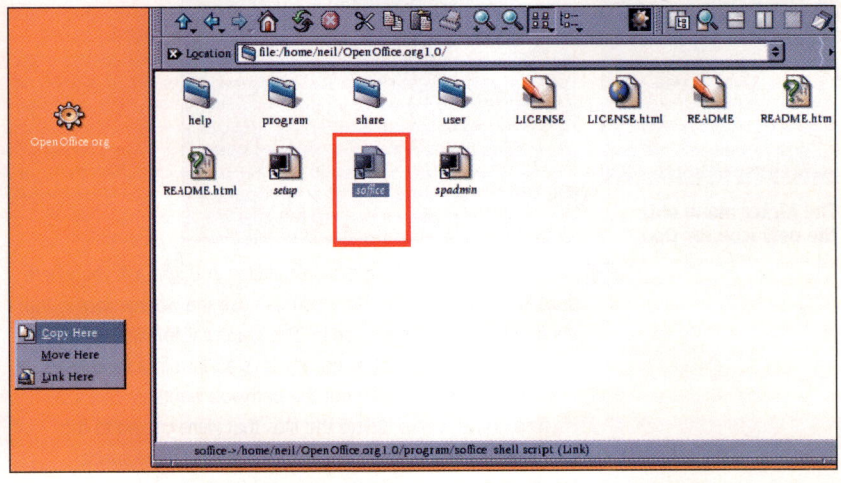
StarOffice was originally produced by Stardivision, a German company, during the 1980s. Sun Microsystems bought the software during 1999 and released it as *StarOffice 5.2* during June 2000. It was a huge suite, including a web browser interface, and it could also make toast and do the ironing. I think. It was really slow too. It came with a calendar (so you could see how slow it was). Nobody really liked the interface, which took over the desktop and served to launch the various parts of the suite.

Sun had another look at the code, adopted an Open Source development model and incorporated many improvements. The saved files are XML based, the file import filters are improved and it was a very different product that Sun released as the *StarOffice 6* and *OpenOffice.org* suites during 2002.

The major difference is that the integrated desktop interface has gone. It no longer acts as a web browser/email client or calendar. The word processor, spreadsheet, drawing package and presentation software are there, along with a lot of other nice touches. The business world uses MS *Office*. Sun's amazing contribution to the Open Source world might very well change that. Both suites open files from *Word*, *PowerPoint* and *Excel*, reliably and sometimes better than Windows opens them, in my experience.

StarOffice is Sun's commercial suite. It's based on the same code as *OOo*, but has lots of proprietary code in it that Sun can not release as Open Source because it does not own the code itself. With *SO* you get lots of time saving templates. *OOo* has most of the major functions (certainly all the vital ones) but you'll have to make the templates yourself.

Having said that, it's free (as a download) so you can't complain. It's supposed to be the largest Open Source project yet.



Drag and drop a copy of the file onto your desktop.

directory with the same name in your Home directory. Inside this are two files that you can link to. The one you want is called 'soffice' but the one called 'spadmin' will also launch the program. Make the 'soffice' file the target of the link. The other way to make an icon on the desktop is to drag and drop a copy of the 'soffice' file onto the desktop. Left-click and hold the mouse button to grab the file, release it over the desktop and choose 'copy here' from the small menu that appears.

It's also nice to have a *Kicker* menu entry. Mandrake provides a utility called *MenuDrake* which allows you to add entries to GNOME/KDE menus – other distributions provide different tools . *OOo* provides an icon (found in OpenOffice.org1.0/ share/icons, called ooo_gulls.png) but I'm going to make my own.

MenuDrake shows a list of icons, but does not tell you where on the filesystem they are found and does not give you an option to navigate to anywhere else, so I'll need to find out where it keeps the icons it can access. First find an icon (from those offered) whose name is easy to remember, then search for the file to find out where it lives. Mandrake keeps the icons for KDE programs in usr/share/icons. That's where I'll put my new icon (or copy the supplied one).

I started *OOo* and took a screenshot of the splash screen as it started up. I trimmed it in *The GIMP* and resized it to 32x32 pixels. This was saved as a .png and copied into the usr/share/icons directory (you might have to be Root to do this). I then used *MenuDrake* to pick up the icon I made. I named the file (*OpenOffice.org* – it's the name that is displayed under the icon) and told *MenuDrake* the path to where the program lives. Now look on your *Kicker* menu and launch the program to prove that it works. If it does not, check that you have typed the path to the program correctly. Of course, this procedure can apply to any program that does not create menu entries and desktop icons.

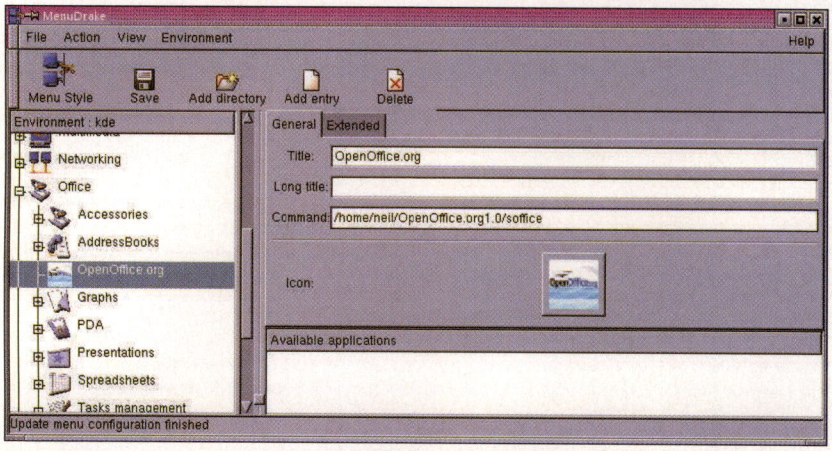
Network setup

The website tells you to unpack the archive (again, use a temporary directory) and install the software in the /opt directory by becoming Root and typing

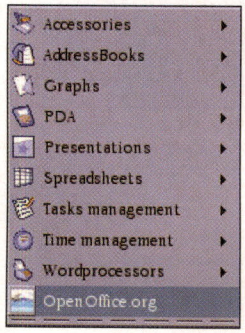
```
/install -prefix=/opt
```

Return to your own user status and navigate to the OOo directory in /opt. Type

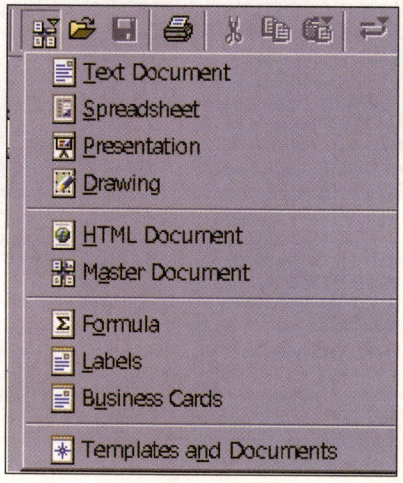
```
./setup
```



Using *MenuDrake* to add an icon to the *Kicker* menu in KDE.



The *Kicker* menu with the new icon for *Ooo*.



The icon to make a new file. You can also use File>New to get here.

and ask it to do a workstation installation. It might ask you where Java lives on your machine – you can use the *whereis* tool in the shell to find the path to type in. The graphical *find files* will also give you results, but not as quickly. It listed Java in a few places, including in mnt/windows; not the path you want!

The last task is to delete the files that were copied to the temporary directory, just to free up some hard drive space. Installation is now complete.

Working with OOo

Launch the suite and you are presented with the word processor. If that's not what you want, you can use the menu: file>new or click the icon (see screenshot) that gives the same options menu. *OOo* will then make a new spreadsheet (or whatever you chose) for you. Before you start using *OOo*, it's worthwhile having a look around the interface and setting a few options to make it easier to work with. Under Help, turn on the extended tool tips. Hold the cursor over an icon and it tells you briefly what it does. (Some programs call this 'bubble help'.) The extended tool tips give a short paragraph telling you how to use the tool. It's a great idea when you are learning new software. Once you know how to use it, you can revert to the standard.

Some *OOo* icons have a small green arrow to tell you that there's a submenu. Hold the mouse button down over it (long click). If you grab the title bar at the top of the menu that appears, it detaches. If you are going to be repeatedly using a couple of tools, leaving the submenus on the workspace as floating toolbars really speed up the work. Some other icons have a long click function but no green arrow. The 'font color' and 'font background' icons change the highlighted text to the displayed colour with a single click. You need to do the long click to change to the colour you want.

OOo keeps the same basic interface but displays icons giving functions appropriate to the kind of document that you are creating, so if you create a spreadsheet, it displays cells in the work area and offers you spreadsheet functions on the toolbars.

What it also does

We will be looking at the major parts of *OOo*: the word processor, spreadsheet and presentation software. *OOo* is a cut down version of *SO*, so you get the same high quality tools that Sun sells as a rival to MS *Office*. What you don't get are all the extras that are in *SO*. What we are going to do now is show some of the unexpected qualities of *OOo*, making a business card to hand out and learning a little about *OOo's* way of working.

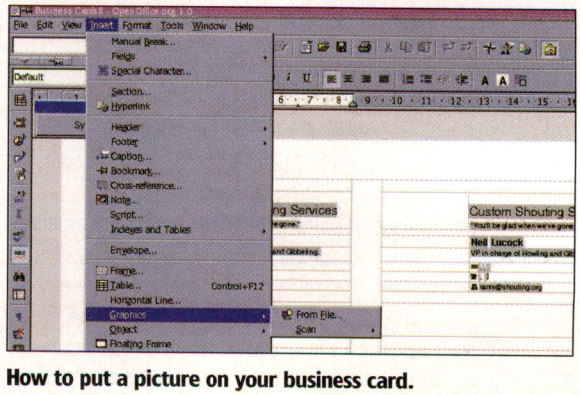
Calling card

OOo can make business cards for you. Nearly every computer shop of any size sells attractive perforated cards in A4 sheets. Buy some that you like the look of. *OOo* includes a nice range of templates to create a suitable card.

First, start a new document and choose 'Business Card'. It gives you a tabbed dialogue box to set all the options. I chose Avery A4 as the Brand, even though the cards I bought were Decadry. It doesn't matter – everyone uses the same size of business card. You'll have to hunt for the right format under 'Type'. Mine were 85mm x 54mm and C2364 seemed to be about right. You have to select one to see the details you need.

Once you think you've got the right size, have a look under 'Format' to make sure that the margins at each side correspond to the margins on the cards you've bought. Go to "Options" to set your printer and tell it to print over the entire page. You can fill in the details that you want on your cards under the 'Business' or 'Private' tabs. Your last stop should be 'Business Cards'. This gives you a preview of what the card will look like. Once you have found one you like, click the 'New Document' button at the bottom. You are returned to the *OOo* interface with all your business cards across the screen.

We now need to make a few changes. I wanted a picture on the cards, so I used the menu Insert>Graphics>From File to choose a picture. The dialogue box gives you a preview. I chose a Mandrake logo (what else is there to shout about except Linux?) and dropped it in the top left hand card. I had not put in a fax or telephone number in the fields under 'Business' but the 'Tel No.' and 'Fax' headings were still displayed. The picture had also moved the URL out of sight. I used the View menu to zoom in on the card and edited it to make it look better. When you've finished, return to the 100% page size. Click the 'Synchronise



How to put a picture on your business card.

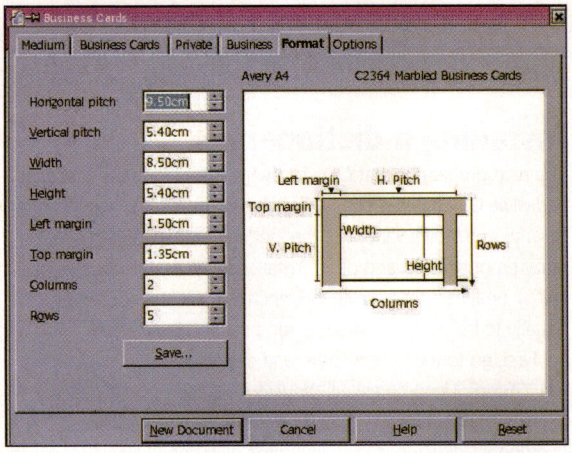
Made In Linux

Blending in to the heterogeneous environment

OOo offers you the chance to get Linux onto the office desktop where you work. Tell people how good Linux is, ask the IT people how they like having to cope with paying for licences for software and also keeping records. I burn Mandrake GPL CDs and offer them to anyone who is interested. Mention that Linux is free (no paperwork is a good argument in every company; high quality and free are even better ones), does everything that a Windows machine does and has no problems with viruses, unauthorised software installations or people 'borrowing' the licensed CDs for their machines at home. It's designed to be secure; compare the number of Windows vulnerabilities on The Register or Slashdot websites to the number of Linux ones.

OOo runs on Windows and Solaris. (The betas for Mac OS X have been released for testing at the time of writing.) You can run it on your existing Windows machines at work (ask permission first) just to show them what you are talking about. Get a couple of people interested in it, ask for their opinion and show them what it can do. I do Railtrack *Powerpoint* presentations at home using *OOo* and email them to myself at work. Mention the fact that it was 'Made in Linux'. Can they tell the difference?

Once a couple of people know how to use *OOo* and have seen what it can do, why not ask if they want to try running it on a Linux machine? Tell everyone you are trying to save the company money. Lots of it. Of course, the IT people might hate you. As long as they have Windows, there'll always be work for them to do.



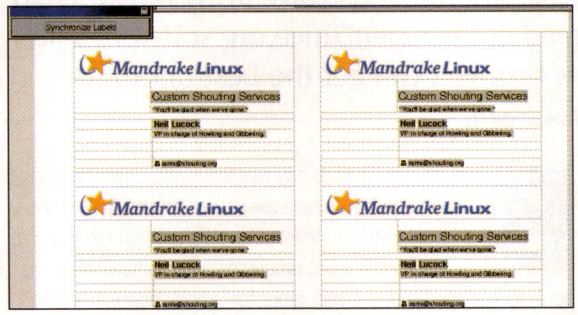
Labels' button and they all look like the one you've just edited. Put the perforated card in the printer and print out your masterpiece. You now want to save the file. When I did this *OOo* put up a light bulb to tell me that the help system had something for me. I clicked it and found it was displaying a page of useful information on the different file formats it saves. If you use the word processor, it will tell you that it has corrected a spelling mistake. They have struck the right balance here.

The Help system tells you it's got something for you, but does not interrupt your work. In many Open Source projects, the programmers make a fine program that works wonderfully, but never get round to writing a decent help system. If you can write, learn the software and volunteer to write help files. It's a great way to give something back to the Linux community.

OOo has a very good help system, comparable to anything you'll find available commercially. I particularly liked the bookmarks idea – it's the first time I've seen that under Linux. If you need to return to a useful page, right-click on that page and add it to your bookmarks. It also allows you to print out the page as a reference. If you want to delete a bookmark, select it in the Navigator window on the left (if it is not displayed, right-clicking gives you the option).

OOo Basic

This was a surprise. *OOo* contains a Basic programming language! There's plenty of information in the Help system. I did not try this out (I'm not a programmer) but you can write macros in the IDE (Integrated Development Environment) that is provided. With the whole thing being modular, you can alter *OOo* to suit your own specific tastes or needs. If you regularly need a particular function for your business, you can have someone write it and integrate it into your desktop.



'Synchronise Labels' makes all the others like the one you were working on.

There's also a developer kit available at the *OOo* website. This is yet another argument for Open Source. Not only do you get more, it also builds on the same principle that ID Software used with *Doom*. Here's the game, here's the tools, now make some levels and release them. It adds value to the software which becomes more capable with time. There have been suggestions that the XML file standard used in *SO* and *OOo* should be adopted as a universal standard.

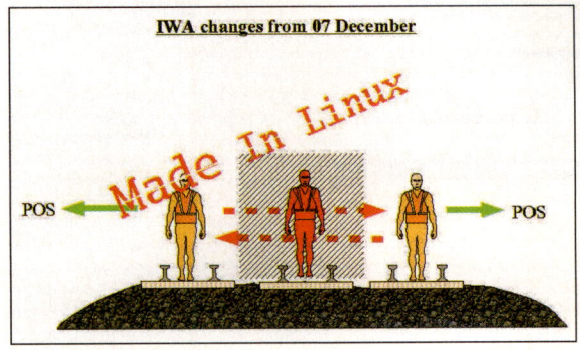
OOo math

If you want custom formulas, there's a nice module with which you can make them.

Conclusions

OOo does have a couple of odd behaviours which should be mentioned. For example, sometimes the menus don't close if you don't choose anything – you have to click somewhere on the desktop to deselect them. Also, if you put one of the 3D objects into a business card as a picture, it does not copy it to the other cards when you try to synchronise. However, the more I use *OOo*, the more I'm impressed with it.

Over the following pages we'll take a look at some of the huge range of features and tools that *OpenOffice.org* offers to the Linux power user. ■



Made in Linux and indistinguishable from *PowerPoint*. This will be used (without the slogan) in Railtrack NW Safety Briefings.

WORD PROCESSING

OpenOffice.org for power users

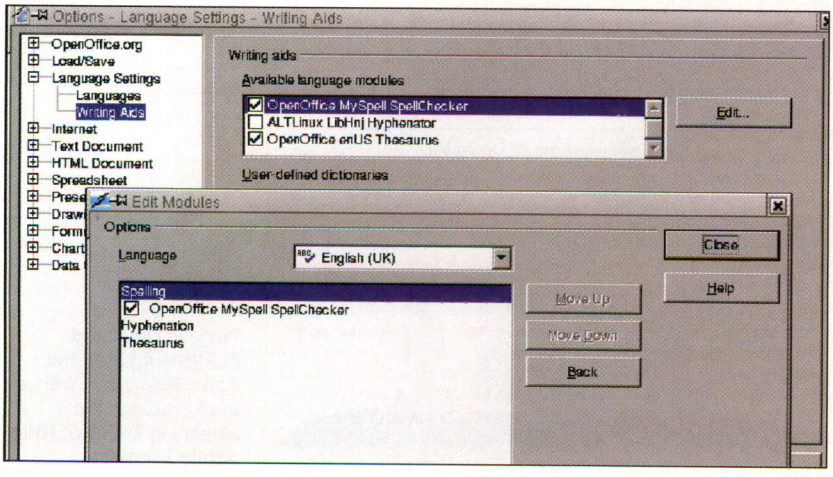
In the first tutorial covering the impressive array of tools and features that are found in the OpenOffice.org software suite, Neil Lucock explains how you can get the best out of its world class word processor.

Before we get down to any real work with *OOo*, I'm going to assume that you have used word processors before, perhaps MS *Word*, *Abiword* or *KWord*. What I'm going to do is look at the word processor part of the suite, show how to set up a dictionary, add more fonts and show you what you need to know to get the most from the best free word processor available for Linux. And I'm writing this article with it...

Spelling

When you launch *OOo*, you are presented with the word processor. (You can change this, if you want to.) The first thing I needed to do was set it up to make learning *OOo* easier. If you look under Tools on the menu or click the icon with 'ABC' and a tick underneath it to the left of the working area, you can ask *OOo* to either spell check a word or to autocorrect. Autocorrect is a useful feature: if you always type "hte" instead of "the", it will save you some effort. However, I am going to type *OOo* a lot in this article, so I need to prevent it "correcting" the two initial capitals.

One way is to go to Tools>Autocorrect/Autoformat and choose Options from the tabbed dialogue. The second tick box down allows you to prevent it from correcting two initial capitals, but that's not the most efficient way of working as we are denying ourselves a useful tool by doing that. If we choose "Exceptions" from the tabbed menu, the lower part of the dialogue box (helpfully called "Words with TWo initial capitals") allows us make a custom rule. I typed *OOo* into the box, told it to add it and now *OOo* just underlines it rather than alters it. The spell checker can be told to ignore all instances of "OOo", but the next time we spell check it, it will ask you for confirmation that the word is correct. Either live with it or add it to the list of correct words.



Installing the dictionary. Once you have unzipped the file into the user/wordbook directory, go to Tools>options and edit the language modules.

Installing a dictionary

The next problem is that *OOo* comes with a US English dictionary and while US English is comprehensible to the British, I would prefer to write in my own variant of the language (despite mad spellings, crazy pronunciation and other irritations that make it so hard to learn). I might want to write in French, so this procedure applies equally to installing a dictionary for a second language.

First, go to the *OOo* website and download the dictionary (en_GB.zip). Uncompress all the files inside the zip file into the user/wordbook directory in *OOo*. Note that there are two "wordbook" directories in *OOo*; there's also one in share. Ignore it. The zip file has a few files inside it. Ensure the original file called "dictionary.lst" is overwritten as the new version has the lines:

```
DICT en US en_US
DICT en GB en_GB
```

That second line tells it to make the GB English dictionary available. You now need to go to Tools>Options and look at Language Settings. There are two options. Choose Languages. If there is a tick by English UK then it will allow you to use it. It probably won't, so go to Writing Aids and edit the available language modules. All you have to do is choose English UK from the options in the drop-down box. When you return to Languages you should find that the GB English option now has a tick by it. Choose it and enjoy correct British spelling. It's not the most obvious method of installing a dictionary, but it does the job.

Now we have got *OOo* using the correct dictionary, we can start looking at the word processor in a bit more detail. Make no mistake, the only light thing about *OOo* is the price – everything you'll ever need is there. It even fills in long words for you (tools>spellcheck>autospellcheck). Type a word and it automatically offers you the rest of the word the second time you start typing it. Press the **Enter** key to accept the suggestion (it doesn't always get it right) or continue typing to finish a different word. It's not a unique feature, but it's nice to have.

What it all does

We need to look around the interface and figure out what all the icons do. When we look at the other parts of the suite in the tutorials following these pages, we'll explain the different icons and sub-menus that *OOo* displays. Most icons will be familiar, but a few have unexpected behaviours.

Some icons have small green arrows to indicate that there is a sub-menu, while with others you have to click and hold to get a

Autopilot

Easy flying

OOo contains an "autopilot" feature. Basically, it's a wizard to help you with common tasks. You select File>AutoPilot and choose, for example, Fax. The process is easy and it gives you loads of options. You can insert information from a database, put in graphics (such as a logo), and a footer message (such as your name and product details). OOo allows you to save the fax you made as a template so that you can reuse the design. You need something like *Efax* or *Hylafax* to allow OOo to send it – it treats it as another printer. Autopilot allows you to make letters, memos, agendas and a few other useful things too.

menu. For example, the text colour icon: click it once and the selected text changes to the last selected colour. Hold it and it shows you all of the colour options available. Some of the menus tear off and stay open, which is a useful property that makes it easier to use a group of tools.

At the top of the program is the usual menu bar and you can get at all of *OOo's* functions from here. Note that the menus are context sensitive. If there's nothing selected, the Edit menu will not give you the options of Cut or Copy. Until you have copied or cut something, you can not see the Paste option under Edit.

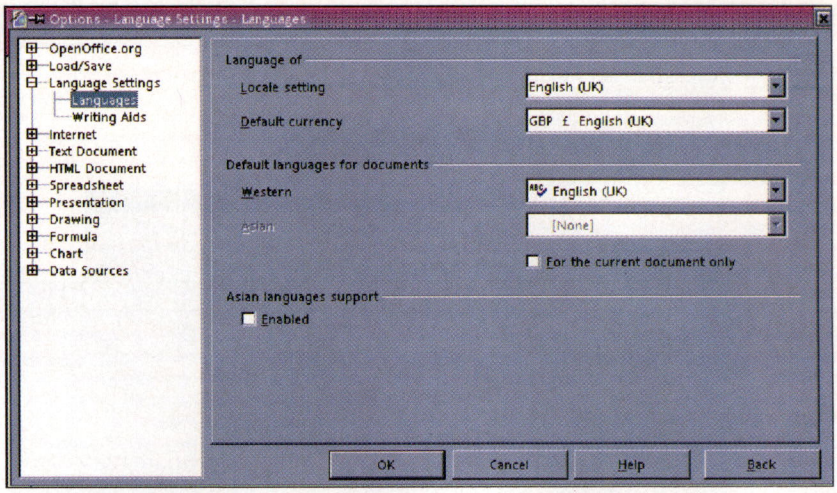
You might want to go to Tools>Options and put in your user data and set a few other options. If you want *OOo* to start as a drawing, do it under the Options>Load/Save>General. You can also tell *OOo* to autosave, but you're using Linux, remember? This program also runs on Windows, which is why the option is there.

Top line icons

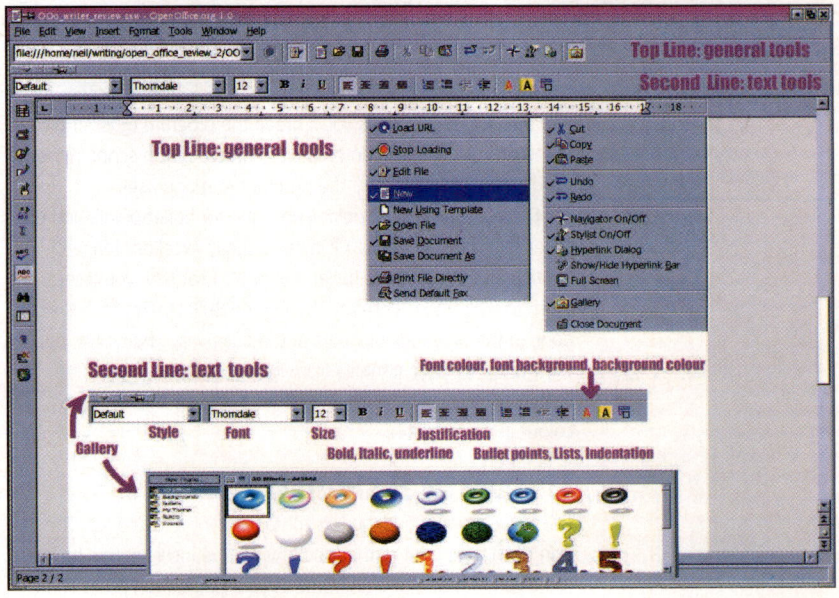
Underneath the menu bar (labelled Top Line in the picture) there is a URL entry field. If you click it open, it shows you the files it has opened previously. You can also type in the filename and path to open a new file. You have a Stop button next to it. I suspect this is left over from *StarOffice 5.2* as it used to include a web browser in the interface. You can stop a file (or website) loading if you feel the need. The Edit File button has odd behaviours. If you click it, it offers to save your changes, then gets rid of the second toolbar at the top and the side toolbar. Clicking again reverts to how it was before you clicked it.

The New file icon opens a new document really quickly. The first time you click it you sit there wondering where all your text went. It's still there; just close the new empty document to get it back. If you click and hold, you get a menu that allows you to choose the type of document to open. The next icons are Open, Save, Print and the Cut, Copy and Paste group. The Undo and Redo icons follow. The Navigator allows you to get around a large document (among other functions). If you place a bookmark (from the Insert menu), name it and then go to somewhere else in the document, you can then click the Navigator button and choose the bookmark you just made. Your cursor jumps to the place where you made the bookmark.

The Stylist applies styles to paragraphs in your document. *OOo* includes a few ready made styles to use but does not display them in the drop-down box. Click Format>styles>catalog and you can use one of the supplied styles or define a new one. You might want all your headings to use an Ariel font at 24 points in red. Just select a paragraph and apply your style to it. The icon next to it is for editing hyperlinks. Finally, there is the icon to start the Gallery. *OOo* includes many of the backgrounds, 3D models and sounds from *StarOffice*. Clicking the icon brings up the gallery browser window, allowing you to choose from a range of graphics to insert. Once you've had a look, click the small arrow at the bottom to hide it. The push-pin icon fixes it in its own window, leaving it open while you write. The icon is a toggle switch – clicking it a second time shuts the gallery down, presumably saving a bit of memory.



This is the second step to install the dictionary. Note that there is now a tick in the "default languages for documents" box. This is the desired end result.



Top and Second lines of icons with The Gallery pasted underneath. It normally appears under the Second Line.

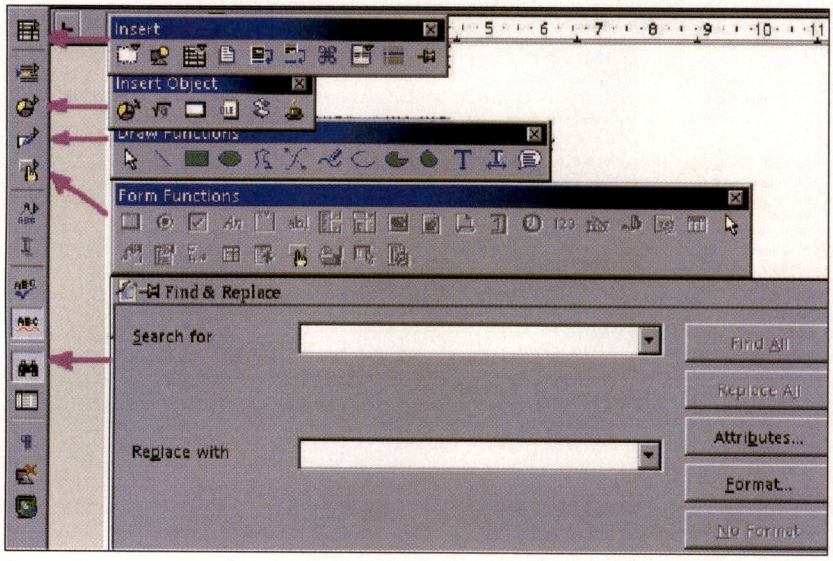
Second line icons

Most of the contents of the second line are obvious to anyone. The Style options is the drop-down box on the left. *OOo* does not come with a lot of fonts, so we might want to make some of the other fonts on our machine available. First find out where they live. If you open "Find files" and look for *ttf you should find a few. KDE keeps some in usr/share/fonts/ttf.

Open a terminal and become root. Navigate to your OpenOffice directory and find the file called *spadmin*. If you attempt to open this from a *Konqueror* window by clicking on it, it just starts the *OOo* program. If you type

```
./spadmin
```

in a terminal window it brings up a dialogue box. It's really for »



The sub-menus on the side toolbar.

printer admin. You use the Fonts button to add some new ones. It copies the fonts unless you tell it to "create soft links only" which saves some disk space. Do not just drop copies of your other fonts into the *OOo* directory as you'll break the program (guess how I found this out...). If you do break it, there's a Setup script in the OOo directory. This allows the program to mend itself.

There are two bullet point icons, one for bullet points and the other for a numbered list. *OOo* has a really excellent range of configuration options for bullet points. You set how you want them to look under Format>Numbering/Bullets. This makes work done in the presentations part of the suite very impressive, but the word processor benefits from having it available too. The button at the far end of the row is for putting a background colour in a paragraph.

The vertical sidebar

The vertical sidebar at the left contains many sub-menus. The top icon duplicates a lot of the functions on the Insert menu. It allows you to put a multi-column frame in your page, then paste some text and some pictures into it. Paste foreign characters or symbols in from here too. Once you use the tool, the icon for that task stays on the sidebar. This is handy if you want to reuse that function (just click it again) but can be confusing if you are trying to learn the menus.

The second one down allows you to paste various fields including the date, time, page numbers, a macro or a hidden message into the text. You can see the hidden messages by clicking View>fields on the menu bar. It's the sort of thing you will either never use or one of those tools you start to use and wondered how you ever managed without it.

Working down the sidebar, the next icon has a pie chart icon. It inserts objects such as charts, applets and the *OOo* Math function creator. Below that we have the drawing functions toolbar. You can instantly start drawing within the word processor.

OOo makes web pages and the next icon down allows you to make forms to use on your web page. You can insert Radio buttons or images, tied into the macro scripts in *OOo*. The Help files have more on this.

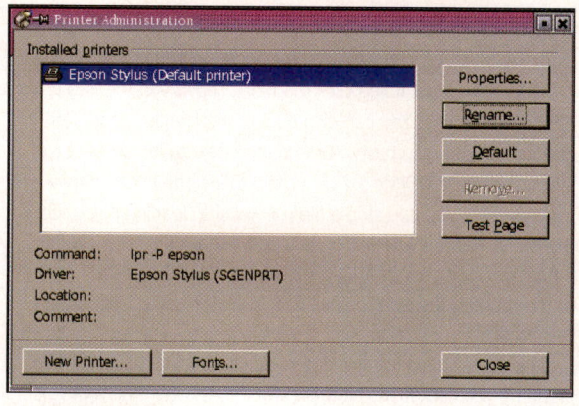
The "Direct Cursor" is activated by the **I** with a drop-shadow. It places a small blue triangle at the left margin. If you move it down a few lines below the last line you typed and start typing, the text appears where you typed, leaving a few lines empty. It also works to accurately place text if you click in the middle of the page and start typing. It puts in carriage returns for you. I've already covered spell checking – the two icons allow you to start a spell check or turn on and off the automatic check.

The binoculars icon brings up the Find & Replace box. This has more functions than other word processors. If you want to find all instances of the word "OOo" and make them appear in a different font and colour, it's very easy. Enter the text you want to find or replace in the top box. Attributes refers to the text already in the document, so if you want to get rid of the bizarre font or colour you used on some words, but leave the ones you changed to Courier alone, you can do it. Type in what you want to replace it with. Format refers to the replacement text, so if you want to replace "OOo" with "Open Office.org" in purple Ariel at 14 points you can. The No Format button removes the Attributes and Format qualities you have defined. Remember that if you don't put something in the "Replace with" box, you'll replace your searched items with nothing. **Ctrl-Z** will undo it.

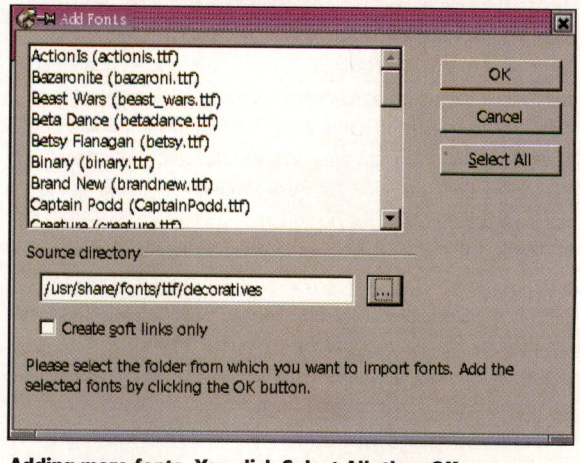
The printer's paragraph mark turns non-printing characters on and off. Word processors use these to format the display on-screen. Occasionally you do something that messes up your carefully formatted page and you cannot seem to get rid of the mistake. Tell it to show all the hidden formatting characters, then delete whatever is causing you grief.

OOo likes memory. If you need to include a few large pictures in your work, it might slow the machine right down and so you can use the next icon down to make it display placeholders rather than pictures.

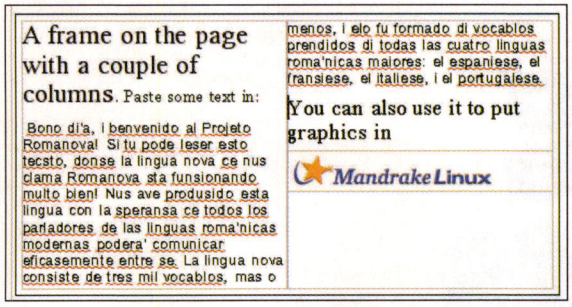
The icon at the bottom of the sidebar shows you what your



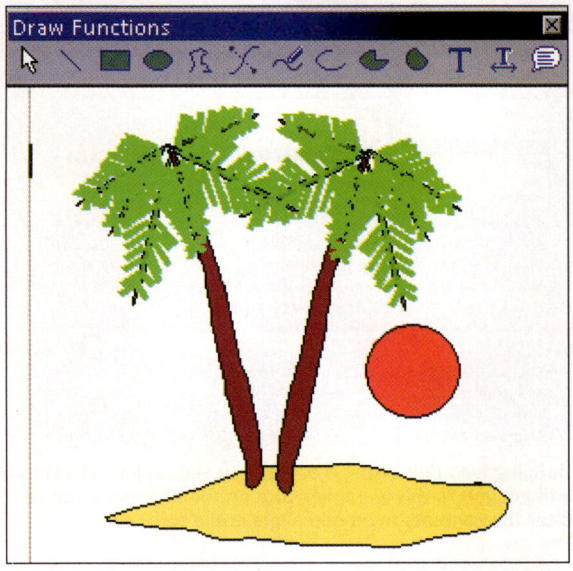
To add new fonts type ./spadmin and you get here. The font button is at the bottom.



Adding more fonts. You click Select All, then OK.


A frame on the page with a couple of columns. Paste some text in:
Bono dì'a, i benvenido al Projeto Romanova! Si tu pode leser esto tecsto, donse la lingua nova ce nus clama Romanova sta funsionando multo bien! Nus ave produsido esta lingua con la speransa ce todos los parladores de las linguas roma'nicas modernas podera' comunicar eficasemente entre se. La lingua nova consiste de tres mil vocablos, mas o menos, i elo fu formado di vocablos prendidos di todas las cuatro linguas roma'nicas maiores: el espaniese, el fransiese, el italiese, i el portugalese.
You can also use it to put graphics in
Mandrake Linux

Multi-columned frame inserted into the page. The top icon on the vertical toolbar allows this.



A quick sketch done directly in *Writer* with the drawing tools.

doc will look like online. *OOo* will happily make web pages, so this might be considered a web preview.

Whenever you are working there is a right-click menu. Once you know what's on it you'll find that it's much quicker to use than moving the mouse to the edge of the screen to click on something. See the screenshot for details of the contents.

Helpful

It is often a valid criticism that while Open Source software is well written, stable and free, you don't always get the most out of the coder's efforts. The people who write code are not always motivated to write Help files or able to explain in layman's terms how their application works. If you don't know what the program can do or can't figure out how to do something, the developers have wasted their time. *OOo* has an excellent and comprehensive Help system. If you use autocorrect it occasionally puts up a discreet light bulb picture in the lower right corner to tell you that it has done something, but unlike some word processors, it's never intrusive and if you ignore it it vanishes after a short time.

The Help files are astounding. You can read them by subject, there's an index with a search function and bookmarks so you can find that useful page again easily. It's the best Help system I've encountered in a free program.

The only criticism I could make is that there could be a few interactive tutorials included to help real beginners (the program includes macros, so it could be done in *OOo* by someone talented) but even without them, the materials provided are well written and always provided satisfactory answers.

MS Word Import

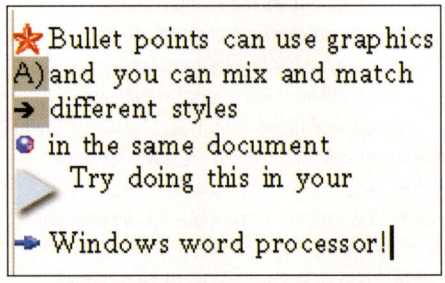
Living with the proprietary world

I almost forgot to mention one of the key reasons to use *OOo*. Tell people that Linux is free and that the office suite is free and they'll be unimpressed. Cost is not the only factor in choosing an office suite. How well something works has to be taken into consideration. Yes, *OOo* might be free, stable and capable, but the office manager still needs to know how to read MS *Word* documents that other people send. Most importantly, can you read the company's existing documents with the new system?

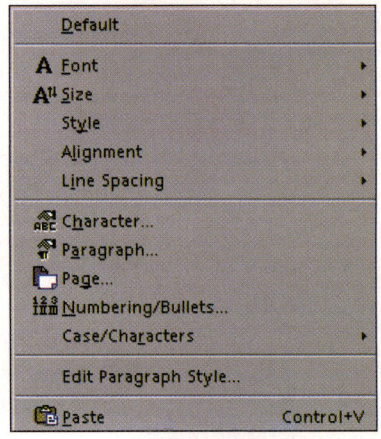
OOo isn't the only Linux office suite to open MS *Word* .docs reliably, but it is the best free one. I quickly tested it on all the MS *Word* docs I could find and it opened them faultlessly, even one I did at work with a map included.

The only minor problem was when someone had used a font that I don't have in Linux and *OOo* substituted another, but it did not look nice (although it didn't look nice in Windows either!). It also opened the MS *Office's* "Contemporary fax.dot" file (the template – see the screen shot) too, so you can still use fax templates that you have a licence for. However, be careful. In some of the new Microsoft end-user licences they forbid you to use their work on other operating systems. Make sure you read licences carefully before accepting them.

OOo runs on Windows, so try a version to see how good it is. Once you have it, think hard if you really have to buy the latest MS *Office*. And once you can open your *Word*, *Excel* and *PowerPoint* documents, your business doesn't need Windows to be productive. What are you waiting for? Save yourself some money and get Linux installed!

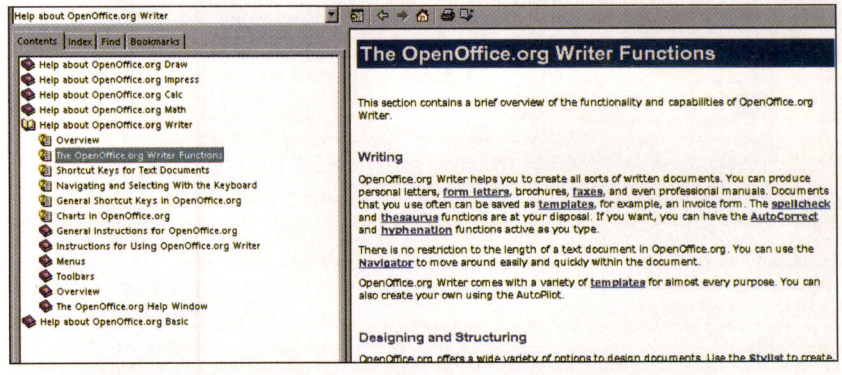

Bullet points can use graphics
and you can mix and match
different styles
in the same document
Try doing this in your
Windows word processor!

The variety of bullet points, all in the same document. Easy!



Right: **The right mouse-click menu. It's quicker than moving the mouse but not as fast as the keyboard shortcuts.**

OOo is big and having looked at it in-depth, I was very impressed. It did everything I could think of and more, plus it worked reliably. It is packed full of features and although I'm wary of saying it is better than MS *Word*, there are certainly a lot of tools in *OOo Writer* that are not in *Word* (compare *Word's* "Find and Replace" with *OOo's*). There were a few quirks though. If you open a menu but don't select anything it leaves the menu open. You have to click somewhere else to close it. When it starts you sometimes have to wait for a few seconds after the document opens before you can edit it, but it's certainly no worse than other programs of a similar size. Get it and learn it and you'll reach the same conclusion as I did – it's a killer office app! ■


Help about OpenOffice.org Writer
Contents
Index
Find
Bookmarks
Help about OpenOffice.org Draw
Help about OpenOffice.org Impress
Help about OpenOffice.org Calc
Help about OpenOffice.org Math
Help about OpenOffice.org Writer
Overview
The OpenOffice.org Writer Functions
Shortcut Keys for Text Documents
Navigating and Selecting With the Keyboard
General Shortcut Keys in OpenOffice.org
Charts in OpenOffice.org
General Instructions for OpenOffice.org
Instructions for Using OpenOffice.org Writer
Menus
Toolbars
Overview
The OpenOffice.org Help Window
Help about OpenOffice.org Basic
The OpenOffice.org Writer Functions
This section contains a brief overview of the functionality and capabilities of OpenOffice.org Writer.
Writing
OpenOffice.org Writer helps you to create all sorts of written documents. You can produce personal letters, form letters, brochures, faxes, and even professional manuals. Documents that you use often can be saved as templates, for example, an invoice form. The spellcheck and thesaurus functions are at your disposal. If you want, you can have the AutoCorrect and hyphenation functions active as you type.
There is no restriction to the length of a text document in OpenOffice.org. You can use the Navigator to move around easily and quickly within the document.
OpenOffice.org Writer comes with a variety of templates for almost every purpose. You can also create your own using the AutoPilot.
Designing and Structuring

The impressive OOo Help system. It's easy to use, complete and intelligent.

SPREADSHEETS

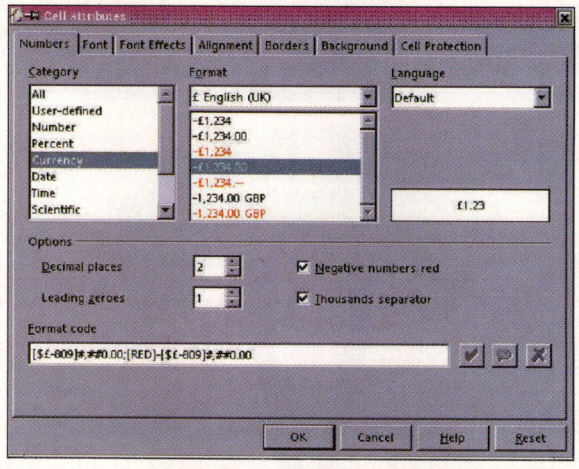
OpenOffice.org for power users

Putting his socks and shoes back on, Neil Lucock attempts to count past 20 using OpenOffice.org's versatile and powerful Calc spreadsheet.

We'll start off with a quick look around the selection of icons that *OpenOffice.org* displays when we start a spreadsheet. *OOo* begins with the word processor, but you can simply go to Tools>Options>Load/Save and choose General if you want to alter this. You get the same basic interface for each part of the suite, it just displays a few different icons (and, in the case of the spreadsheet, the cells) depending on what type of document you make. *OOo* is easy to get used to – most of it stays the same no matter what you are doing. However, don't let the simplicity and price make you think *OOo* isn't up to the job – it is a professional product with a comprehensive range of tools. If you can use *Excel*, you can do the same job in *OOo Calc*. You'll just save money.

Toolbars

OOo gives you a couple of toolbars to access useful commands. Toolbar 1 has a currency format on the left. Spreadsheets allow you to set defaults for how data entered into each cell is displayed. Normally the default is General – it displays numbers as they are. You can highlight a cell (or a Row, Column or group of cells) and change how the data is treated. Currency settings are changed under Format>Cells... by choosing Numbers from the tabbed box. *OOo* supports the Euro as well as the original European (and non-European) currencies. You can also define how you want your money to be displayed, so if you want a couple of leading zeros before the decimal point if the amount is less than a pound, you configure it there.



Defining how *Calc* displays any money values. Format>Cells will get you to this dialogue. Click on the Alignment tab to alter the contents to an odd angle in the cell.

You can also mandate a variety of options from the same dialogue box. *OOo* benefits from a high degree of integration of the different parts of the suite. You can easily make the contents of a cell display at 90 degrees to the rest of the sheet if you need to. This isn't a vital feature, but it's nice to have.

Useful

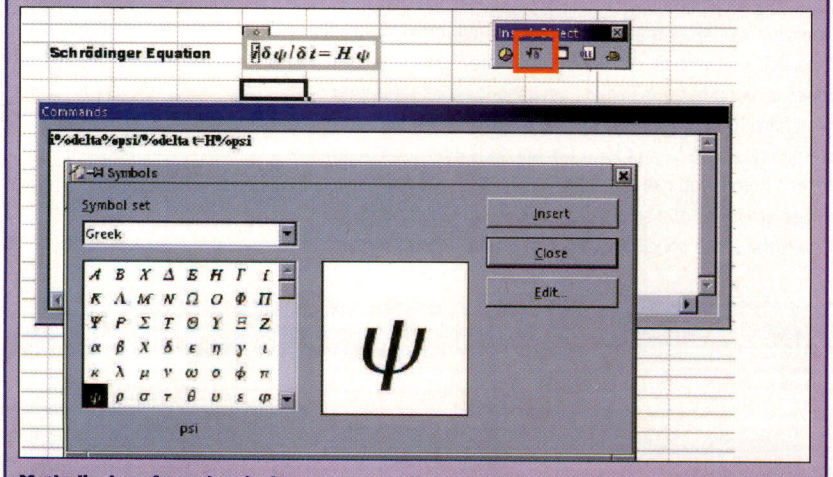
The next icon shows a percentage sign. If you type 1.23 into a cell and click the icon, you'll get 123% and typing 0.6 gives 60%.

The third icon, a dollar sign, a percentage sign and a backwards arrow underneath, removes the formatting the two previous icons imposed. It's a sort of specialised undo tool. The next two icons add and remove decimal points to the number entered in the cell.

The rest of Toolbar 1's icons are for cell formatting. You can move the contents to the left or right within a cell, put borders around all or just part of it and change the background colour. If you have a deep cell (perhaps because you have inserted a picture which has made all the cells on that Row wide enough to fit it) you might want to alter the vertical display position of your

Math

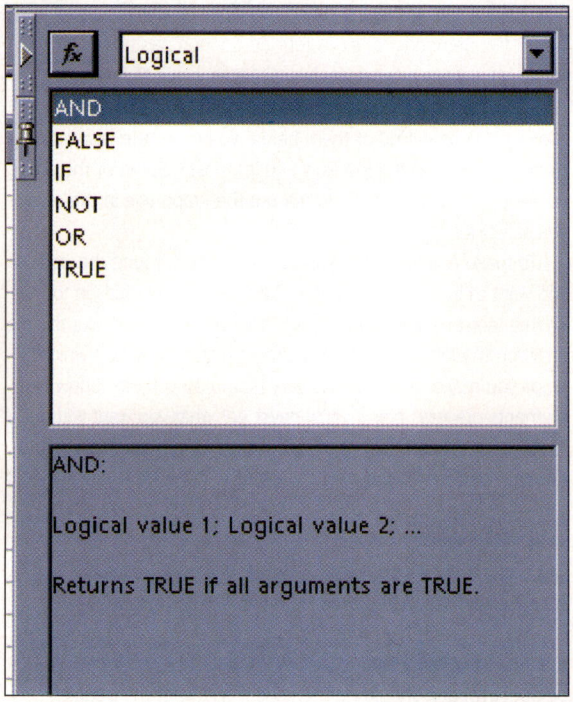
Displaying formulas



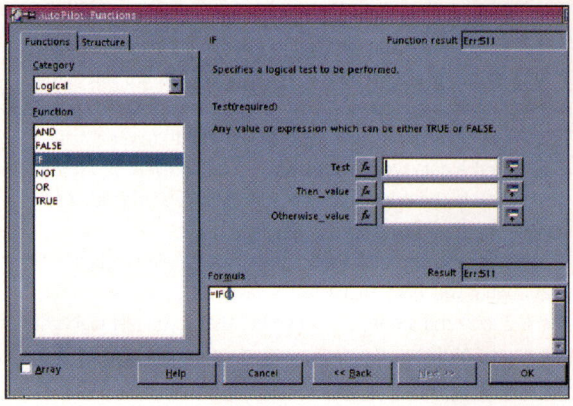
Math displays formulas, it does not work them out. That's your job.

OOo includes *Math*, a program to insert formulas into your spreadsheet. Click on the icon with a ? on it. *Math* then changes the toolbar and displays a box to allow you to enter your equation.

Note that this is not a utility whose purpose is to calculate the formula shown – it is actually a way of displaying the formula, nothing more. This utility is really just a specialist word processor applet, which extends the usefulness of *OOo* if you have to reproduce complex formulas for publication.



The Function quick reference gallery, usually hidden over on the far right of the spreadsheet. The types are usefully categorised and there's a reminder on how to use each one.



The Functions Autopilot (wizard), accessed through the Insert>Function menu.

figures. The last three icons on Toolbar 1 let you move the contents to the top, middle or bottom.

All cells in a spreadsheet have a reference, with letters across the top and numbers down the sides. If you need to go beyond Z it uses AA, AB, then BA, BB and so on. The cell you are working on is displayed on the left of Toolbar 2.

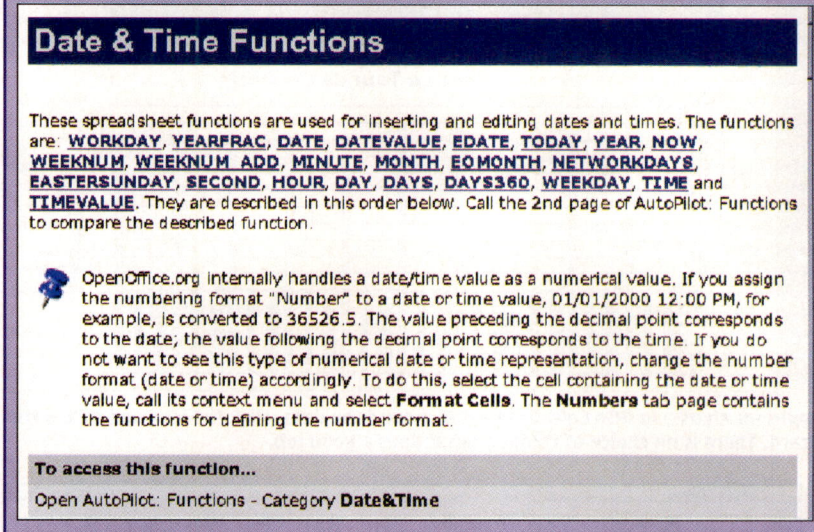
The icon that looks like a keyboard launches the Function autopilot. This tool offers help in creating various formulas. You need to know what you are doing mathematically – it won't make you into Einstein, but if you need to work out inverse hyperbolic cotangents it will help you to get the data in the right order.

SUM thing

The **SUM** character is next. It adds up the list of numbers above it (by default) so make a few numbers in a column, click in the cell below them and it adds them all up and displays the result. If you want it to add up a Row, click in the Cell where you want your answer, then click and drag over the numbers you want to **SUM**. Hit the **Enter** key to confirm. You can display a **SUM** anywhere

The Help System

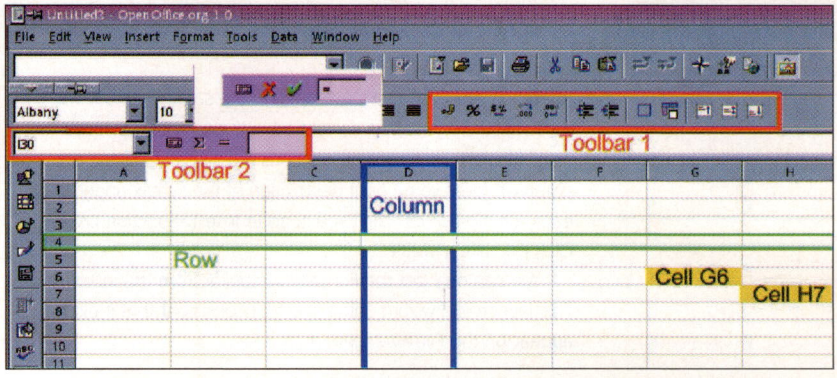
An unobtrusive friend


Date & Time Functions
These spreadsheet functions are used for inserting and editing dates and times. The functions are: WORKDAY, YEARFRAC, DATE, DATEVALUE, EDATE, TODAY, YEAR, NOW, WEEKNUM, WEEKNUM_ADD, MINUTE, MONTH, EOMONTH, NETWORKDAYS, EASTERSUNDAY, SECOND, HOUR, DAY, DAYS, DAYS360, WEEKDAY, TIME and TIMEVALUE. They are described in this order below. Call the 2nd page of AutoPilot: Functions to compare the described function.
OpenOffice.org internally handles a date/time value as a numerical value. If you assign the numbering format "Number" to a date or time value, 01/01/2000 12:00 PM, for example, is converted to 36526.5. The value preceding the decimal point corresponds to the date; the value following the decimal point corresponds to the time. If you do not want to see this type of numerical date or time representation, change the number format (date or time) accordingly. To do this, select the cell containing the date or time value, call its context menu and select Format Cells. The Numbers tab page contains the functions for defining the number format.
To access this function...
Open AutoPilot: Functions - Category Date&Time

A page from the Help files. The picture shows you the kind of detail you get with *OOo*. There are plenty of hyperlinks included, the system is easy to use and you get some nice explanations.

OOo has a comprehensive Help system that is well written and easy to use. It is searchable, you can place bookmarks in it and it has a good success rate. Congratulations to whoever wrote the intelligent behaviour. Unlike the intrusive Help in MS *Office*, *OOo* flashes a small light bulb picture in the bottom right of your display. It is telling you that it has either done something for you (such as autocorrected a mistake) or it thinks that you might like to know something. It offers you help on file extensions when you first try to save your file, but it is never intrusive. Ignore it and it goes away after a short time.

The entries are comprehensive and the only thing I would like to see is a few tutorials. However, *OOo* is so easy to use it's a minor issue. The level of detail is good and I'd say that it's the best Help system I've found in a Linux program.



The *OOo* interface. The alternative Toolbar 2 display is also shown. Refer to the article for explanations of the various icons.

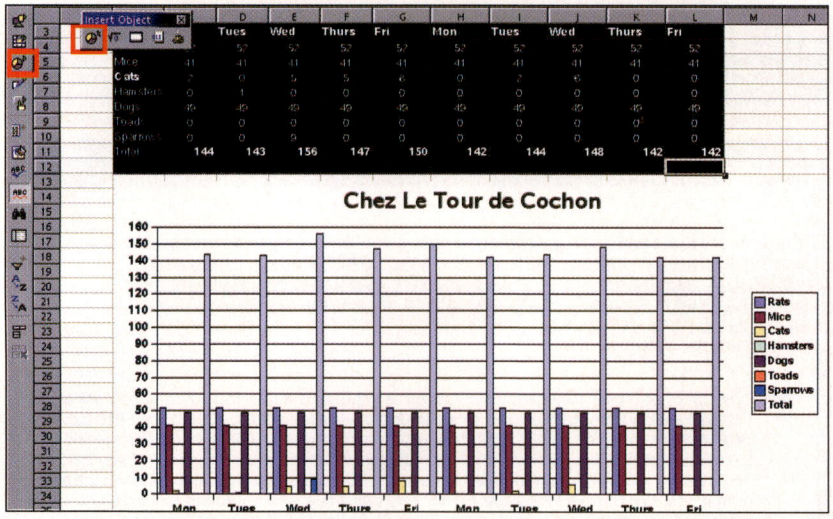
you like; it does not have to act on the cells next to it. When you use **SUM** (or other formulas) it changes the Toolbar to show a tick or cross. The tick (or **Enter**) will insert your formula while the cross abandons the operation.

Formula

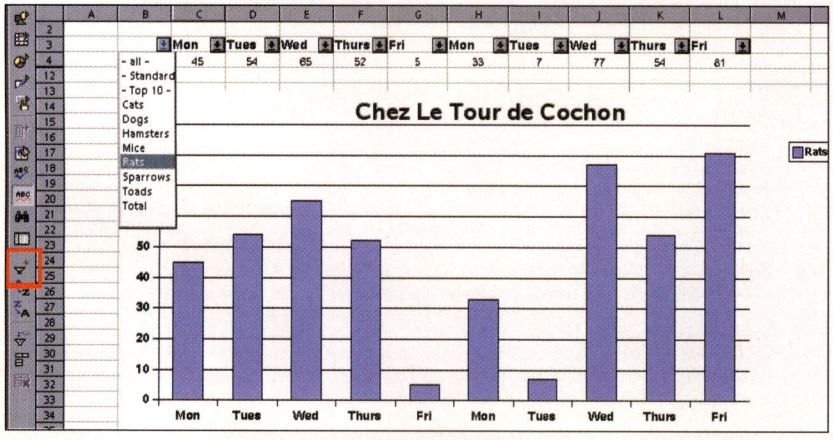
The equals sign is clicked to tell the spreadsheet that you want it to treat the thing you are typing as a formula and then act on it. If you type a command without the equals sign, it will display it as text, which might be what you want to do. The box to the right of the equals sign shows what the formula in a cell is and you can edit it here. The cell in the body of the spreadsheet shows the value of the formula.

Over on the far right of the work area is an unobtrusive arrow and push pin. This opens out to act as a formula "quick

»



Graphs (or charts) in *OOo Calc*. Drag a box around the data, click the icon and follow the wizard. There is no choice of colours, but it does a good job.



The Auto filter can cause just one type of data to be displayed.

« reference" that you can leave open in the same way that you can with The Gallery (shown concealed in the screenshot above the word Albany). The types of functions are sensibly divided into groups. It gives you a brief description of the syntax underneath each function. If you can't see it, go to Insert>Function List and click to make it visible.

You can only get the Function Autopilot from the Insert menu. This launches a wizard and it seems inconsistent that it did not appear on the File>Autopilot menu.

Graphs (or charts)

OOo makes nice graphs and it's not difficult to alter their appearance. You can change how they look, move the graph around and, by right-clicking, get an Edit menu. One quality I liked was that if you click on a bar in the graph, the cell it refers to on the spreadsheet is highlighted. However, if the graph you made isn't what you wanted, there's not a lot you can do. Just delete the graph and recreate it.

Note that if you change any of the values that have gone into the graph, it updates to show the latest data. Sometimes the Auto Calculate behaviour is not what you want. If you have a really big spreadsheet and need to make a lot of changes, it can be annoying to have to wait for everything to be recalculated. Tools>Cell Contents>Auto Calculate toggles it on and off. F9 will then recalculate the sheet when you are ready.

I have a spreadsheet showing the types of animals eaten each day at the notorious "Chez le Tour de Cochon" restaurant. Although it looks nicer if you leave spaces in the body of your data, *OOo* will put the spaces into your graph as "Row 4" or "Column B", so get rid of them before you make the graph. *OOo* uses the names in the top Row and leftmost Column of the area you select as data names. Do not put the graph title in the area you select for the graph.

To make a graph, first click and drag a box around the data you want to include, and get the Row and Column titles in too. Then either use the "Insert>Chart" menu item or click out the pie chart on the left hand side. The program starts a wizard which helps you make the graph. It's very helpful to tick the "Show text elements preview" box – you'll never get what you want without it. You can alter the titles and put a heading in before telling it to Create. You then drag out a box and *OOo* draws your graph.

3D or 2D?

OOo can create various types of graph, although some are not very useful. Think hard about what you are trying to show and choose the graph type that shows the information in the clearest manner. 3D graphs tend to distort the data (they include perspective) and should not be used if you want the people reading the graph to have an accurate understanding of the relationships shown on it.

The Auto filter icon is worth investigating if you use graphs. Perhaps we want to show just the number of rats eaten each day at our restaurant. It can be used to select certain data values from a database shown on the sheet for display. We select the Rats option in the drop-down box the graph updates to show us just the number of rats eaten. Be careful not to turn off Auto Filter when a data type is displayed – it can prevent you from going back to the original display. It's probably a good idea to ensure you have saved your graph first.

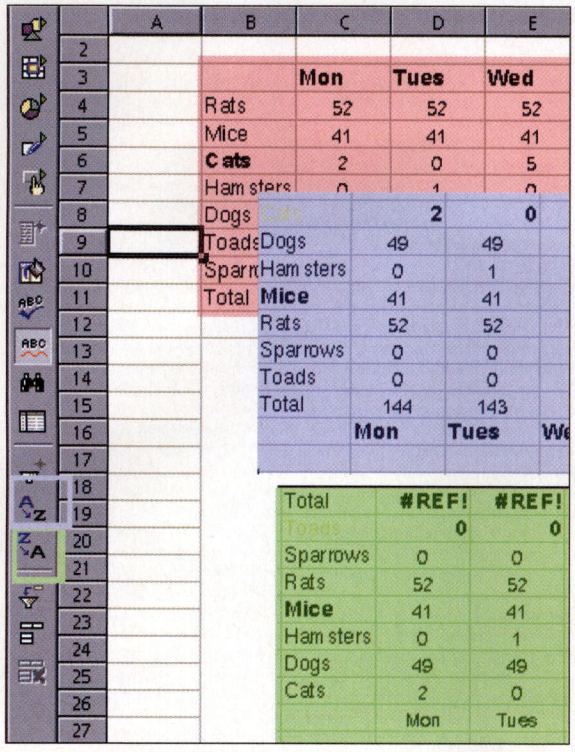
We can also sort alphabetically, or reverse alphabetically if that is what you want. Highlight the range of cells to sort, click the A-Z icon and it does it. Remember that if you just highlight the names of things, the data alongside is left unchanged. If you highlight the whole chart, it sorts the names and the data that belongs to the names, but be careful that cells within it that perform any kind of mathematical operation on the contents are not affected. It's best to make sure that the totals, averages and so on are not highlighted.

Gnumeric, ergo SUM

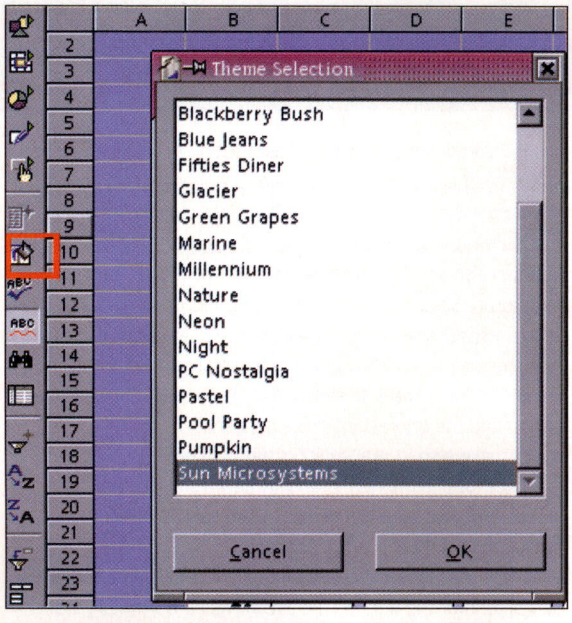
Sun comes through

I had a problem with the SUM feature when I first tested it. I made a simple spreadsheet, saved it as an *Excel* file and imported it into *OOo*. I got an error as a result of this because *OOo* did not like how *Gnumeric* saved the file. I was not sure where the problem was, but Jonathan Mills of Sun UK (which sells *StarOffice*, the commercial version of the code), contacted me via *Linux Format* and asked for the spreadsheet and looked into it. It seems that the problem lies in *Gnumeric's* code somewhere – there's actually nothing wrong with *OOo's* SUM functions.

What does that tell you about the attitude of the people at Sun who distribute the code? Would Microsoft contact someone who thought they had found a bug in their code or just deny it existed? Welcome to the world of Linux, a genuinely better way of using computers. Many thanks to Sun UK for helping to sort out where the problem was.



Be careful with the alphabetic sort tools. The red shading shows the original data. The blue is the ascending filter, with the days of the week moved to the bottom. The data is still okay, as "Total" came after "toads". The green area shows any errors which are found at the top since the cells no longer add up correctly.



Theme selection. Note how the colour of the cells has changed to mid-blue.

More useful tools

However, people make mistakes, alter things or paste parts of the spreadsheet into other areas and then wonder why it no longer works. Sometimes you forget what you are trying to do. *OOo* provides a tool called Detective to help sort out problems.

Highlight the cell where you get an error, click on Tools>Detective and either Trace Dependents or Trace Precedents. A red arrow or blue outline box appears to show you which cells contributed data to the mistake. *Excel* has a similar feature (called Auditing) so it's not original, but if you are familiar with *Excel*, you'll feel right at home.

The Themes icon (a paint bucket) allows you to apply a colour scheme to your spreadsheet. *OOo* gives you a few to get going. It's not something you might think of a use for, but if you have lots of spreadsheets to deal with, it's not a bad idea to make the backgrounds of all one type of spreadsheet (for bonuses earned, for example) all one colour as it might help prevent mistakes.

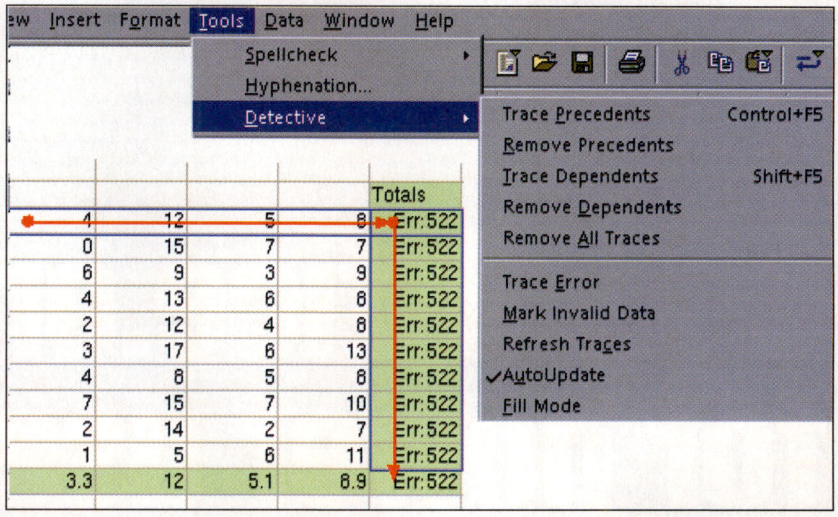
Imports and Exports

The Linux balance of payments

OOo **will open and save your spreadsheets in Microsoft *Excel 97/2000/XP* file formats. I have a few *Excel* spreadsheets on my Windows partition. Whenever I try to open a spreadsheet with *OOo*, it always opens it correctly. The only minor problem you might have is that sometimes you don't have the correct font on your Linux machine. I've tried every file I have and never encountered any problems, but that does not mean that you won't have problems. Load *OOo* (it also runs within Windows) and try it first.**

Data is precious, and so you should ensure that it loads the files you use. If you have custom spreadsheets at work, try them first, compare the data produced by a Windows *Excel* sheet to the same sheet in Windows *OOo*. If it works okay in the Windows version, you should have no worries in Linux.

This means that *OOo* could well be the Linux "killer application". Now you don't have to use Windows to open your *Word*, *PowerPoint* and *Excel* files.



The Detective tool, which is useful for finding errors in a spreadsheet. Click the cell to find out what other cells contribute data to it.

Conclusions

OOo Calc is a deep program and you could easily fill a whole magazine on this one subject – there's so much to it that it's hard to decide what to leave out. You can make interactive forms (there's a wizard to help you), use Java or OLE objects. I could spend pages discussing Functions alone, but the Help files should give you enough advice.

If you want a replacement for *Excel*, *OOo Calc* has all the tools you need. If you need to reduce costs for software licences, *OOo* is free so you can install it on as many machines as you please. If you are worried that the company that provides your present software is just treating you as a source of income and you don't really want to have all your data tied up in a deliberately incompatible format, then *OOo* could be the solution you need.

I haven't used the Windows version of *OOo*, but if I relied on selling MS *Office* for a living, I certainly wouldn't be taking out any new bank loans... ■

PRESENTATIONS

OpenOffice.org for power users

In this OpenOffice.org masterclass, Neil Lucock tries to give an illustrated talk on "What I did on my holidays" while upside down at 5,000 feet. If the typing appears upside down, please invert the magazine.

OpenOffice.org's presentation software, *Impress*, was the first thing I ever used in the suite. It's competing with Microsoft's *PowerPoint*, the industry standard, so I'll be considering how well it compares and whether it offers a viable alternative. As always, I'll presume that you have a working version of *OpenOffice.org* on your PC, know how to open and save files and make directories.

Getting started

OOo is a genuinely integrated office suite. *Impress* benefits as you can use the tools provided in *Draw*. I'll cover using those tools in detail in the next tutorial (page 32), but I'll need to discuss a few here in order to make a presentation. Launch *OOo* and choose File>New>Presentation to start *Impress*. The autopilot (wizard) starts by offering to either open an existing presentation, use a template (although *OOo 1.0* doesn't include any, you can save your own) or create a new one. The next dialogue asks for your slide design. Again there are no options offered. *OOo* is the Open Source version of *StarOffice 6* (*SO*) – the templates and designs are in the commercial product.

You can define how the finished document will be used: as slides, overhead projector transparencies, paper or as an on-screen presentation. Next, define the transition from one slide to the following one if you are creating an on-screen presentation. You can also set the display times if you want your presentation to autorun. Set it to manual (the default) for the time being as you can easily alter it later. The final step in the wizard is to define how the slide looks. They give you a useful selection of 20 layouts to use. I prefer to start with a blank slide. Click the Create button and the *Impress* interface appears.

I'll make a short presentation and we'll have a look at the essential tools in *Impress*. I recently flew a small plane and did some aerobatics, and I'll use this as the subject of my presentation. I think I'll need six slides.

The first slide is going to be the title. I've chosen a military stencil font for the titles as the aircraft was a US military trainer. There's a "T" icon on the left tool bar near the top which lets you put a text box in your presentation. Most of *OOo*'s icons in *Impress* have a sub-menu that you can bring out by clicking and holding the left mouse button down. The small green arrowhead tells you that it's there. Once the sub menu is displayed, you can make it stay open by clicking on its title bar. You can then reuse a tool without having to find it in a menu. I draw a lot in my presentations and I find this feature makes *Impress* easy to use.

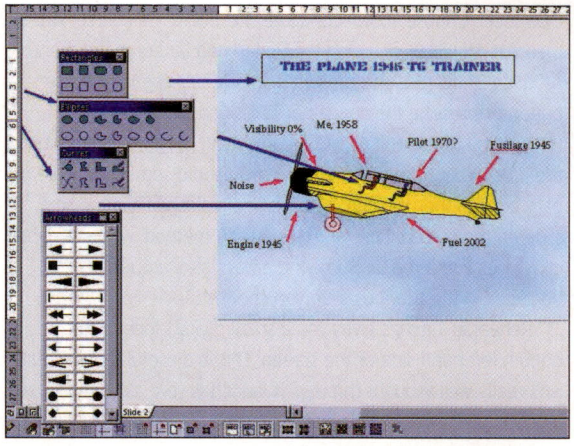
There are a couple of other useful tools on the text toolbar. The "T in a box" scales your text to the size of the box you drag. If you change the size of the box, the text inside it automatically

re-scales itself to fit the new box size. The "Callout" (which looks like a speech balloon) is for labelling things. Click the item you want to label, drag the line out to where you want the details to appear and it gives you an empty box. Click the T on the text sub-menu and you get an insert point. Type in the details you want to display.

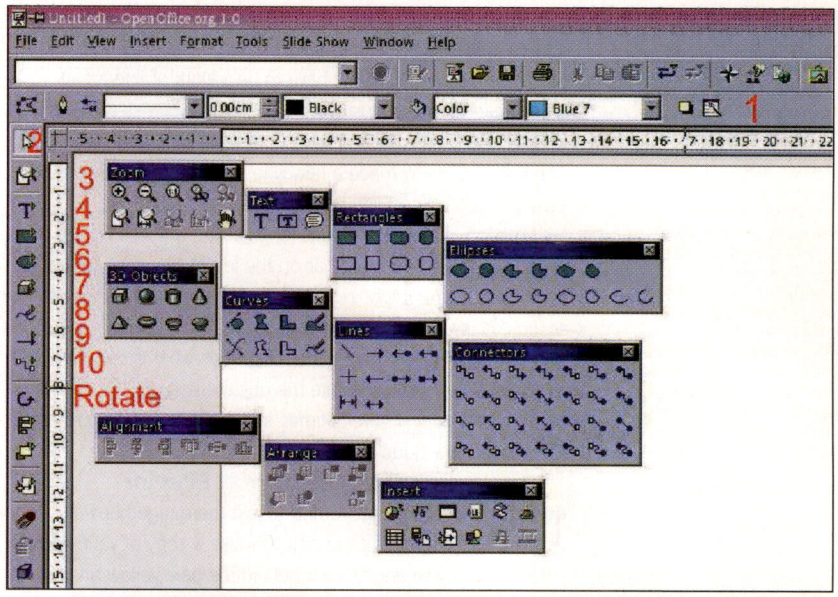
Picture this

I've put a picture of the plane in (Insert>Graphics) and resized it. The clouds background is from The Gallery. *OOo* includes the backgrounds, fancy bullet points and 3D objects that are in *SO*. You launch it with the icon that looks like a framed picture to the extreme right of the top line of icons. Just find a picture you like and drag and drop it. A hint: put the background in last, otherwise it becomes difficult to select text and any other pictures.

When you drop your background onto your slide, it covers up everything. About a third of the way up from the bottom of the left hand tool bar is a useful series of icons for positioning. We need to send the background image to the rear, so it is behind everything else. Each item (either a picture, drawing or text) on the presentation slide can be compared to a picture on a sheet of transparent plastic. Newly made items are nearer to you (on their own sheet of plastic) and will cover items made previously. We have the option to send things backwards or bring them forwards. Click on an item to select it, then click on the icon in the sub-menu to send it to the back. There's also an easier way. Right click after selecting the object and one of the options is Arrange. Choose what you want to do from the options offered.



Pilots are a series of ellipses. Note that *OOo's* spell checker is active, underlining fuselage. The plane was traced from the photo. It you can recognise it as a plane, it's good enough.



The left side sub-menus. 1: This area changes according to what is selected. 2: The select arrow. 3: Zoom tools. It includes a tool to make the display the true size, *e.g.* when printed. 4: Text tools. 5: Filled and unfilled rectangles. 6: Ellipses. The tool on the far right of the top line was used to make the clouds. 7: 3D objects – rotate them or alter the lighting, to be covered in the next tutorial. 8: Bezier curves and shapes. Really part of *Draw*. 9: Lines. 10: Connectors. 11: The rotate tool. Select something, then click on this to interactively turn it. Under Rotate we have Alignment, which is useful if you want to align objects relative to the top, side or bottom of the slide. Arrange brings an object in front of another or puts it behind. Insert puts graphics, spreadsheets, photographs and formulas into your presentation.

I want the presentation to be light-hearted rather than serious, so I've added an arrow in red and the word "AAAAAGH!"

For the second slide, I drew the plane using the drawing tools, then put a note to everything I want to talk about. I know that when I come to the one that says "Engine 1945" I'll talk about the noise, the vibration and the fact that you can't see in front of the plane when it's on the ground. "Fuel 2002" reminds me of the quote from the company's website that the plane is a very efficient method of turning gasoline into noise. These notes mean nothing to anyone else but will prompt me to talk about each item. It's like having a secret list that no one else can read. You cover all the items in your talk because they are in front of your »

Different Views

For good preparation

Impress shows you the Drawing View as a default. (View>Master View>Drawing View). This allows you to edit the slides and make new ones. The other options are as follows:

OUTLINE VIEW: This allows you an overview of your show. Clicking the small icon to the left of the preview shows you what is on each slide. You can rearrange them by drag and drop.

SLIDES VIEW: If you did not set a general rule for slide transitions when you started *Impress*, you can define how each slide appears, the speed it arrives at and (if you have the display set to automatic) how long each slide displays for before moving to the next image. There's a timer you can (in theory) use to practice getting the length right and an icon to hide slides. Perhaps you have prepared a presentation but realise that the person who spoke before you has already covered some points. Just temporarily hide the slides that you don't need in your presentation. Right-click for a menu to include it again.

NOTES VIEW: Notes have three functions. Firstly, they remind you exactly what it was that you were going to say before nerves, distractions or just plain ignorance left you standing in front of a crowd of people without a clue as to what you are supposed to say next. They are a backup for the occasional mental segfault. Secondly, they can be printed and distributed to the audience after your presentation. It saves them taking notes. Thirdly, they are a useful backup if your equipment breaks or is otherwise unavailable. If you carry your notes, you can still do your presentation. Photocopy your notes, hand them to the audience and refer to the pictures on each sheet. Good presenters always have a "Plan B" in case something goes wrong.

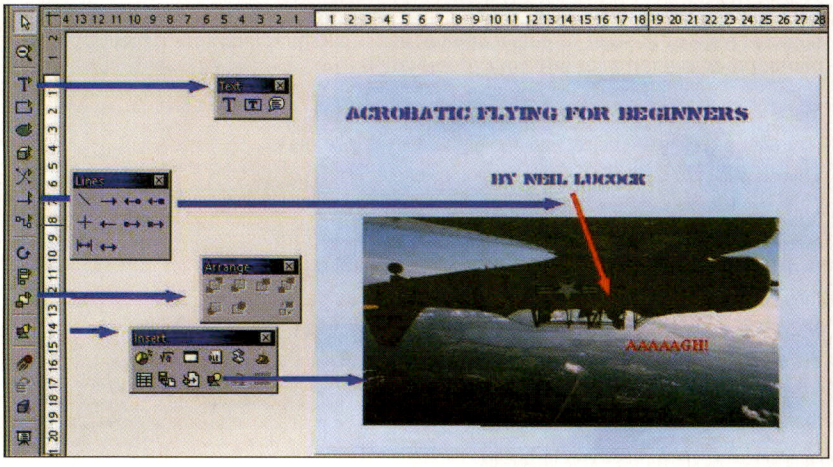
HANDOUTS VIEW: *OOo* makes handouts with your pictures on. If you want to add your details, a few arrows or lines, you have all the usual *OOo* tools available. I put a simple text box at the top, then made a new hatching type and put that over one of the slides that I don't plan to show.

eyes. The plane was drawn by importing a picture, drawing around it, then deleting the picture to leave the outline. Tidy it up and add a few details. It doesn't have to be a work of fine art – presentation drawings just need to be simple and clear. As long as people recognise what you have drawn then it is good enough. Use Insert>Slide to made a new one after the one you are editing. Edit>Delete Slide gets rid of it.

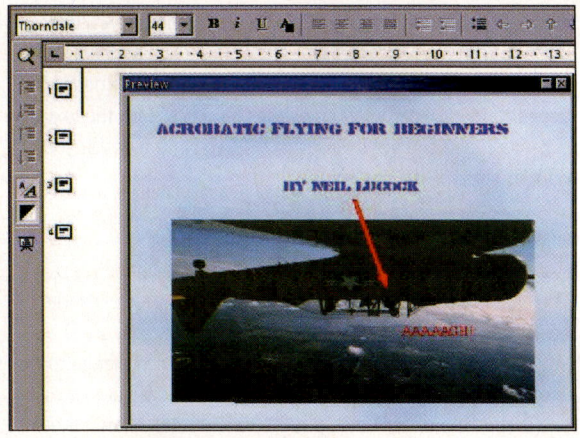
I made some simple clouds for the slide with the flight path on it. There's an Ellipse toolbar on the left of the work area and an Ellipse Segment tool in that. It produces a circle with a flat section. Make one, make the line around it invisible (the tools above the work area change depending on what is selected) and fill it with a pale yellow. Rotate the object using the Rotate tool on the left. Now copy it several times and make the copies different sizes to make a cloud shape.

Drag a selection box around them all. Either use **Ctrl+Shift+G** or Format>Group to join them together. Pale yellow doesn't stand out too well so make a copy of your cloud, colour it pale grey and place it behind the pale yellow one.

Sometimes you need to include large images in your presentation and they slow down your machine as you are working on it. Perhaps your presentation will be printed out on monochrome printers as a handout and you want to make sure that the details are still there without colour. *Impress* can display everything in colour (the default) but you can have it display as a greyscale or in black and white. View>Display Quality allows you to change how it appears. Note that this change is not permanent – you can put the colour back in again.



You can make an acceptable presentation using just a few of *Impress*'s tools. I've shown exactly what I used for this slide.



The Outline View enables you to drag and drop your slides into a different order.

Making Slide Shows

On the menu bar there's a Slide Show heading. *OOo* has the usual keyboard shortcuts (**Ctrl-C**, **Ctrl-V** and so on). It says that **Ctrl-F2** runs the slide show but in fact it just drops the program onto the KDE taskbar. You can configure your own shortcuts for *OOo* tools. Choose Tools>Configure>Keyboard. Of course, **Alt-S**, then **Alt-W** will work, using the underlined letters in the menus.

Slide Show runs the presentation. Clicking the left mouse button or pressing the space bar takes you to the next slide (unless you have set it to run automatically). Right-clicking during the presentation wasn't consistent as it would sometimes take me back a slide, then at other times it briefly flashed the previous slide on the screen. I prefer *PowerPoint's* method, where a right-click gives you a menu. Someone always asks a question that begins, "On the previous slide you showed...".

Rehearse Timings gives you a small digital clock on the bottom left hand side of the screen. This is a useful feature that isn't really well thought through. It launches your slide show, you practice saying what you need to say and figure out how it will all

Doing a Presentation?

Get the basics right

So you have to stand in front of an audience and talk about something? I'm not going to discuss presentations in detail, but I need to address some issues in order for you to use *Impress* effectively.

The first thing to do with any talk, speech or tutorial is to plan it. That involves finding a pencil and some paper and writing down what subjects you need to cover in the order you think best. Most presentations benefit from a heading showing the subject, who is speaking and what topics are to be covered.

It's a good idea to do a summary too.

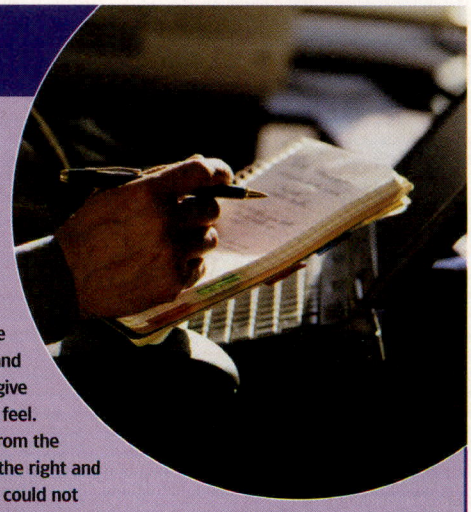
A lot of people seem to forget that presentation software is just a tool. They think that it will do the work for them if they just use the supplied wizards to make their presentation. They then wonder why their audience is acting bored while they are reading it at them. Treat your software as you would a blackboard. If you need to show something, it can display an image for you. You can then talk about the image you have displayed, explaining it.

Yes, they can take it in quickly, but they are not going to be getting ahead of you. You remain the focus and it remains the tool.

A final point: presentation software often comes with flashy slide transitions. You can make one slide dissolve, slide in from one side or another, or include supposedly amusing noises.

These detract from what you are talking about. Is the intention to show what the software can do or to inform the audience? If you really must use them, consider how a written document would look with lots of different fonts all together.

Make all your slide transitions come from the left hand side and it will give it an integrated feel. Bring some in from the left, the top or the right and it looks like you could not decide what to do, and that's certainly not the impression you want to make.



fit into the time you have available. Unfortunately, it doesn't give you a summary. What you need is something that says "Total time – eight minutes" or similar. Both *PowerPoint* and *Hancom Office's* presentation software do it that way. *OOo's* version displays the timer so small that you've got to be sitting right next to your screen. You'll have to be next to a flat surface anyway as you'll need a pen and paper to write down the time each slide needed and then you'll have to add it all up to find the time it took. It would be much easier to find a clock and note the start and end times when you practise. The tool really needs more work before it is any use.

If you've ever done a loop in a plane, the worst part is not being vertical or inverted – it's the bottom of the power dive which you need to do to build up speed before you pull up. All the blood flows to your stomach and legs and the G-forces make it uncomfortable to breathe. I've drawn a loop and labelled parts of it. I intend to ask the displayed question, get responses from the audience and then display the answer. To do this, I used Insert> Duplicate Slide to make an exact copy of the previous slide. I then deleted everything except the answer. Click to highlight the text box and the hatched area. Clicking Slide Show> Effects will enable you to assign an effect to your text and objects. I made the text appear on a mouse click, shooting in from the left one letter at a time.

Slide Show>Interaction is a nice tool. It allows you to set special behaviours when you click an object on the page. You can set it to go to a different slide, make a noise or run a program. It's almost like having hypertext links included in your presentation. If you wanted, you could quite easily use a presentation as an interactive teaching aid.

Conclusions

Most of the tools in *PowerPoint* are present, but there are some things which are missing. For example, you don't get the ability to convert files to Adobe PDF, probably for copyright reasons.

Impress is a good piece of software in its own right. It has a good set of tools and while it is not a clone of the Microsoft program, you should have no trouble making the transition. There are a few things that could be better (the timer, as an example) but there's nothing left out that you would need. Perhaps the best argument in its favour is the cost. Linux is either a free download or a cheap CD. *OOo* is the same. If your business has just ten registered copies of MS *Office*, running on ten Windows PCs, you are probably looking at £3,000 to £4,000 in software that will be obsolete in a couple of years. You'll be asked to constantly upgrade, and that will cost you even more while *OOo* will still be free. Buy one CD (not more than £5 from the place you get your Linux CDs from) and you can install it on all your PCs and you don't have to worry about record keeping or the BSA knocking on your door. How much does the licence record keeping (to prove that you have bought licences for all the copies of your software) cost you? How much would a prosecution harm your business? *OOo* and Linux have benefits that go way beyond the cost of the software itself, and I didn't even mention the reliability of Linux once. ■

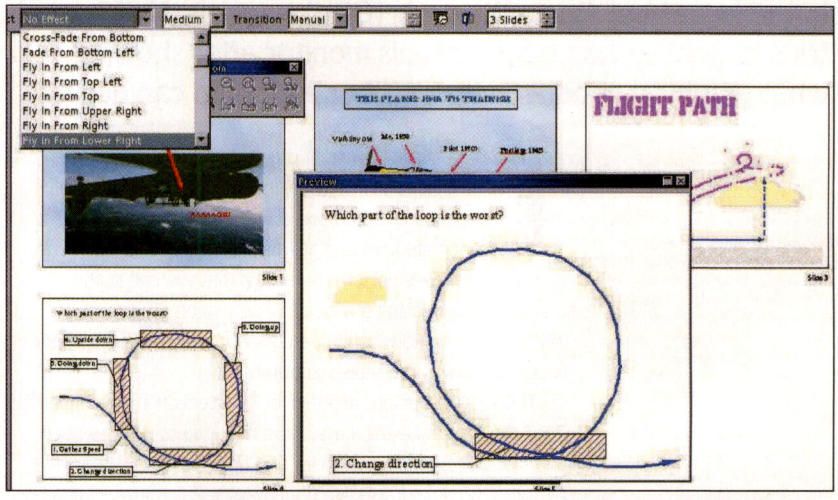
Notes View. If your mind goes blank when you see your audience, have everything written down already.

PowerPoint Import and Comparisons

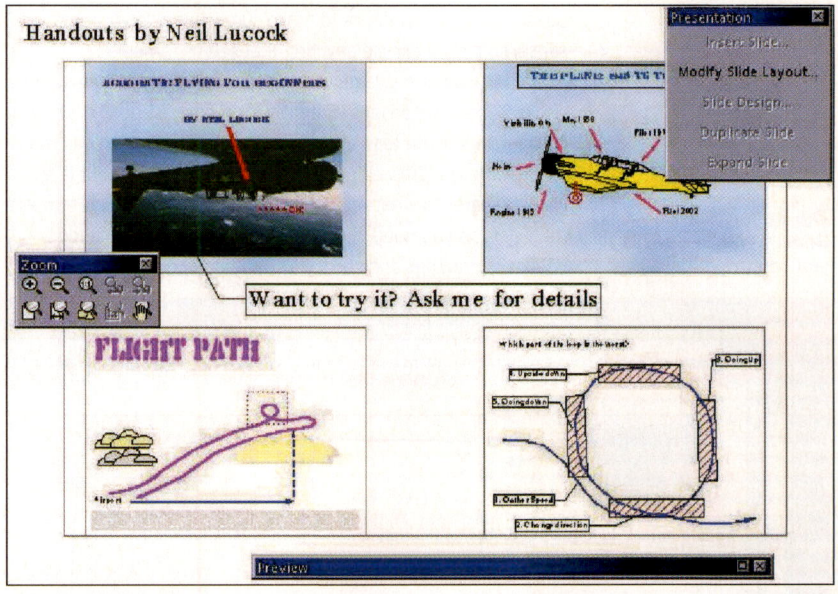
Living with the opposition

When *Linux Format* did their reviews of Linux office suites, *OOo* and *SO* showed that there was a viable alternative to MS *Office*. *Impress* will usually import *PowerPoint* materials created with anything up to *Office 2000*. Unless the particular font isn't on your machine, you won't tell the difference. It exports reliably too – I've drawn maps in *Impress* and used them with *PowerPoint* at work.

However, as with any Open Source program, don't expect it to be 100% effective, particularly when it comes to importing files from new software releases. I was sent some *PowerPoint* Safety Training presentations. I'm not sure what version of *Office* they were made with, but while *Office 2000* opened them, *OOo* failed. It takes a while for someone to write import filters for the latest version.



Slides View. If you did not set the way your presentation displays when the program started, you can set various effects and hide slides from this display.



The Handouts display allows you to insert notes and captions.

DRAWING

OpenOffice.org for power users

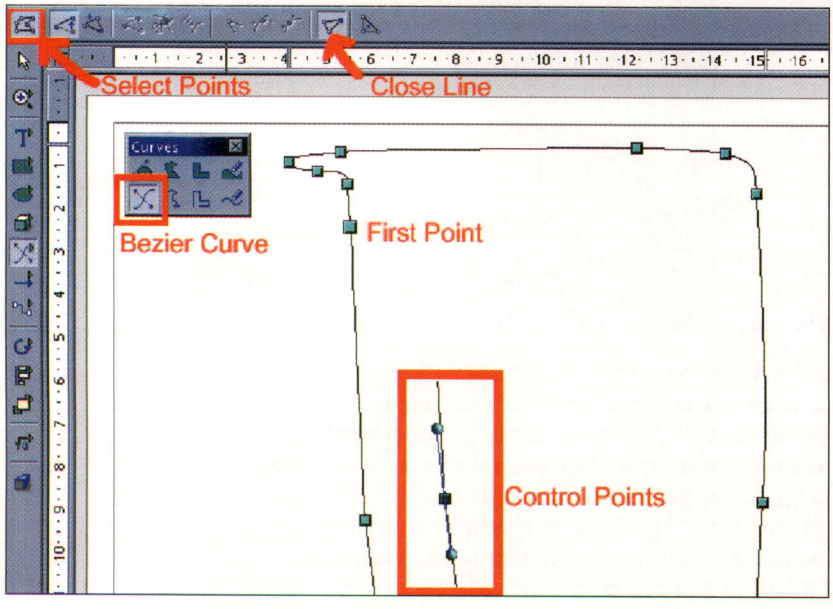
In the last of our OpenOffice.org (OOo) tutorials, Neil Lucock tries to get the wax crayon off his monitor after showing what graphical goodies Linux's killer application can deliver.

There are two types of drawing programs in common use. *The GIMP* and *Photoshop* are bitmap editors – they manipulate coloured pixels. *Draw*, *Freehand* and *Corel Draw* create lines and filled shapes mathematically. The advantage of these vector programs is that you can scale a drawing without losing any details. Zoom in on a bitmap (or a JPEG) and eventually you'll see coloured squares. Zoom in on vector artwork and it scales smoothly.

This tutorial was put together in *OOo* version 1 running in KDE 2.2.2. on Mandrake 8.2. I assume that you know how to open and close files, make directories and hold a mouse. However, vector drawing packages are not always very intuitive – you click on something and apparently nothing happens. Although not intended to be an introduction to vector drawing techniques, we'll have to cover a few basics to make sure what follows makes sense, so please excuse anything that seems remedial.

In our look at the features of *OpenOffice.org* in this magazine, we've made comparisons with MS *Office*, as most people who use computers at work tend to be familiar with the programs in that suite. There is a drawing applet available in MS *Office*, but it in no way compares to what you can do in *OOo Draw*. We'll go through the key tools you need to produce your own illustrations, do a couple of drawings and show you what it can do.

The Bezier tools. Understand these and you can draw anything.



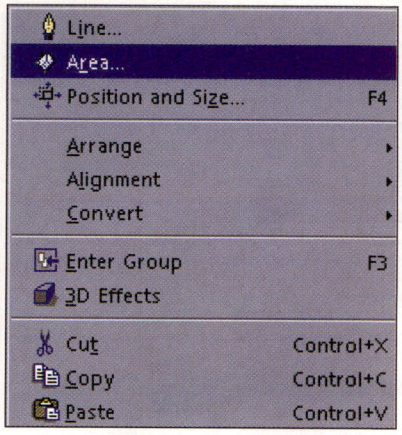
Start *OOo* and click File>New>Drawing. *OOo* gives you a blank drawing area. I'm going to draw the view in front of me (a couple of computers and a monitor). It helps if you think of an object as a series of shapes. First, I want to draw the coloured panel on the front of my Linux PC. It's roughly rectangular and the temptation is to find the rectangle tool. Not a good idea – once a rectangle, always a rectangle. The best way is to use the Bezier tool (found under Curves – see the boxout). Click the tool to select it, put your mouse cursor over where you want the shape to start (it doesn't have to be exact since you can change it later) and click and hold the left button as you move the mouse. If you just click and release, you'll get nothing.

Select Points

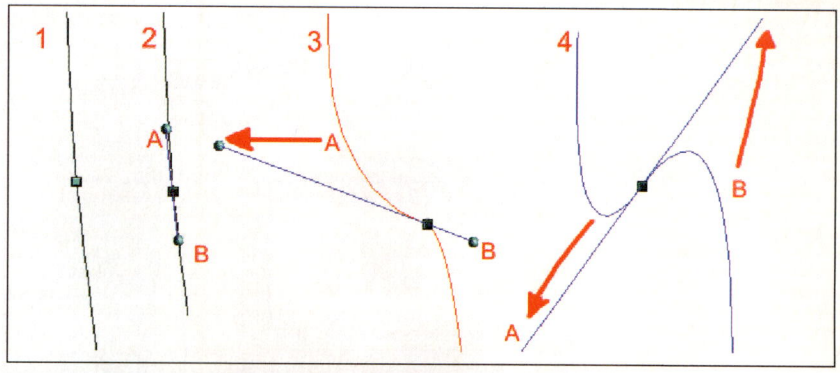
It should start to draw a line and it will continue drawing the line until you double-click to end it. Move the mouse and click to put more of your line in, particularly where there is a change of direction. What you are doing when you click is putting Control Points in, and you can use these to modify the final shape. Draw a shape that's fairly close to the real thing and end the line. Click the line to select it and small green boxes appear around the shape. You then click on the Select Points icon. There are two – one at the top of the left side toolbar and one at the bottom; use either. These change the line into a series of Control Points. There's one at each place you clicked when you made the shape. When you click the Select Points icon, the top toolbar changes to give you some useful point editing tools. The one I need here is Close Line, which joins up the two ends. This will eventually allow you to fill the inside with colour. Before you do that though, you will need to get the shape right.

Each Control Point can be selected and moved. The line between it and its neighbours stretches to the new position. You can alter the shape of the lines on each side of the Control Point by clicking on it a second time. Two balls appear on screen, attached to the Control Point by short lines. The First Control Point and the last one you made only have one ball to move. Moving these balls further away from the Control Point affects the adjacent line over a greater distance. Moving them at right angles to the drawn line (in other words, away from it) alters the curvature. Sometimes the two balls are directly over the Control Point. Grab one and move it, then grab the other. Understand this behaviour and you have the key to making anything with a vector program. Move your Control Points as you wish until you have the shape you want, then click a second time on one of the Select Points icons to finish editing.

We made this a closed shape so we could fill it. Select your shape and right-click to bring up a menu and choose Area. You can also use Format>Area from the menu. This dialogue box is



The right-click menu. Different entries appear, depending on what is selected before you right-click.



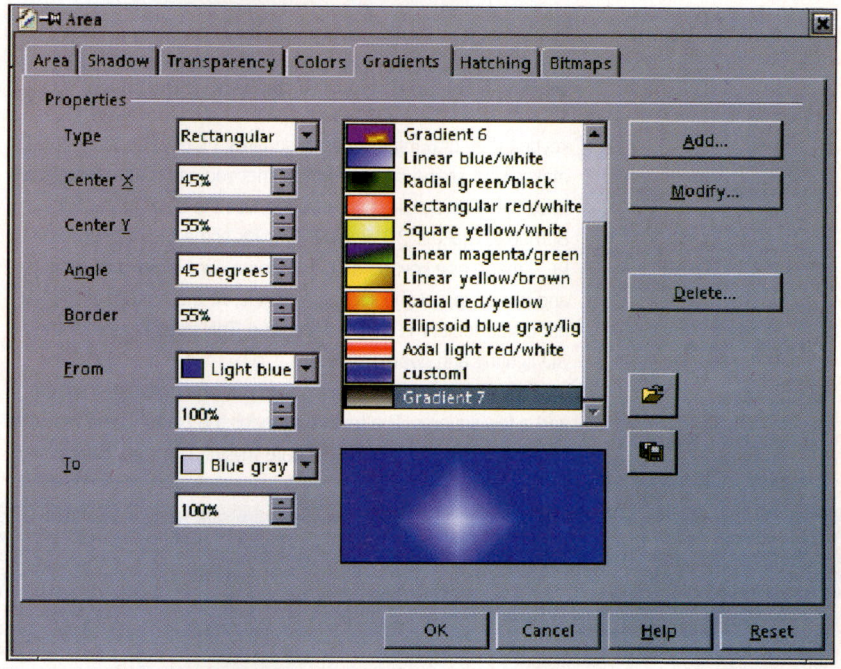
1: A Control Point. 2: Click it a second time to show the control balls. 3 and 4: The results of moving the balls around – the lines change shape.

another major tool for making your art look good. There's a tabbed dialogue box. Choose Area and you get a choice of various fill types.

Shadow works like the dropped shadow you can assign to text. It puts another slightly offset copy of the shape behind the original. You can set the offset to fall in one of eight directions, you can make the shadow any colour you choose and you can adjust the transparency. Higher values make it more transparent. Transparency can be set for the whole shape or as a gradient. The gradient has options to set the kind of gradient that you require and is very configurable.

Colors allows you to choose from the 90 shades already in the program or make new ones. Gradients and Hatching both allow you to make new designs and use them in other parts of the suite. You can assign a bitmap to your shape and there are tools to adjust its position and size. I've used 3D programs that did not have such easy to use tools.

I coloured the shape blue, then used the same technique to make the case, the details and the smaller BSD machine that lives on top of the Linux machine. Once you have positioned parts, it's a good idea to group them. Select a part, hold the Shift key and select another. Then use the menu Modify>Group or right-click for a menu. Once grouped, the parts of your fledgling masterpiece can be moved and resized. You can always use Ungroup if you need to alter something.



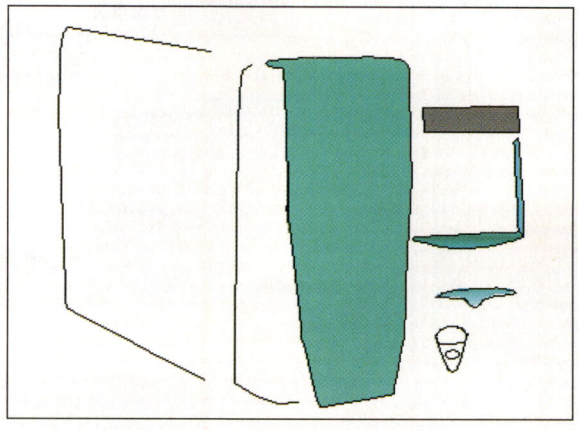
The gradients dialogue, accessed by right-clicking and selecting Area.

Into the third dimension!

The monitor started as a series of grey rectangles. I made the screen by taking a screenshot with *KSnapshot*, then pasting it in with Insert>Graphics. The lower part of the monitor has a pattern. I used Format>Area and made some new Hatching. This was applied to two boxes and the lines around the outside of them were made invisible. The tool to do this appears directly above the working area whenever you select a line or pre-made shape. I then drew another line around the outside of both boxes.

I made a small box and made the contents the same hatching type I had used for the bottom part of the monitor, but with a transparent background instead of grey. Place it in front of the other part and it looks like the speakers. The manufacturer's names on the monitor and smaller PC were done with the "T in a box" text tool. This scales the text in the box when you scale the box. It's a good tool to use when you want to include text in your drawing but the font size won't go that small.

The illuminated button at the front was made with *OOo Draw*'s 3D tools. The primitives provided can be modified (so, for example, a sphere can have the number of segments increased or reduced). You can change the lighting, the shadows and the

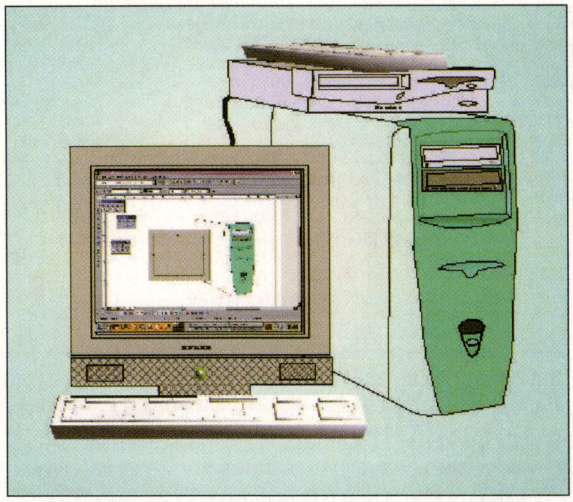


Making parts to create the larger PC.

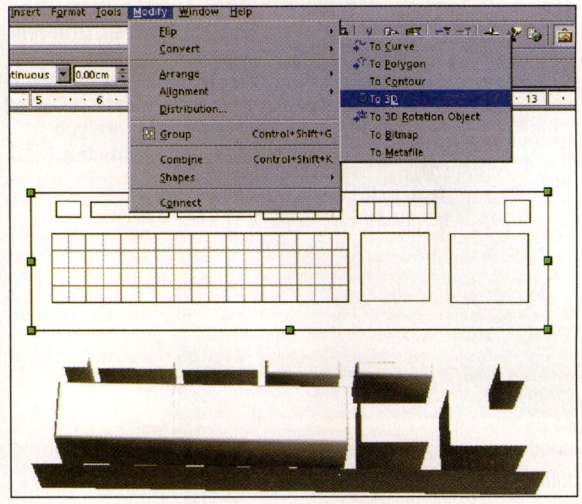
colour of your model. Okay, it's not *3DS Max* – you are not going to do the graphics for a game in it – but it's nice to have.

The switch is an illuminated sphere with a grey sphere in front of it. Select the shape you want from the 3D Objects toolbar, then click and drag it onto the work area. Once you have made it, you can use the Rotate tool to move it in three dimensions. Perhaps circles and squares are not much use to you, but *OOo Draw* can make 2D drawings into 3D objects! I made the

»



The finished picture. The background is made simply by sending a coloured square to the back of the image.



The series of boxes that were used to make a 3D keyboard at the top of the page and the result below it.

keyboard by drawing a series of squares. Modify>Convert>To 3D makes your shape into something that you can rotate. Once you have the object in the right position, use Modify>Convert>To Bitmap to make it 2D again. This saves some memory on your PC. I made one keyboard, then made a copy which was rotated and put on top of the second machine. At the bottom of the left-hand side toolbar is a 3D Effects icon. This gives control of the 3D lighting and geometry.

I also made a small space scene out of a pyramid, a sphere and a few squares. The texture map for the planet came from the Net and I drew the ship hull texture myself. The stars background was mapped onto a square and I placed it behind the other objects. Remember that *OOo Draw* isn't a true 3D program – It's a drawing program with some nice extras. Even with half a gigabyte of memory, the screen display became very slow as the viewpoint was scrolled when looking at the drawing of the PCs. The same applied when moving the 3D objects.

I used the Text tool to write "Space is Dark" across my picture, changed the font to a pseudo-Japanese style and used the Format>Fontwork tools to bend the letters into a semi-circle. The Fontwork tools allow you to put your text along a curved line, add shadows and other attractive effects.

Guide to the OOo Draw Toolbars and interface

Expandable sub-menus that stay open to help you work

Across the top the default is the Line tools. The Edit Points tool is associated with Bezier tools, mentioned opposite.

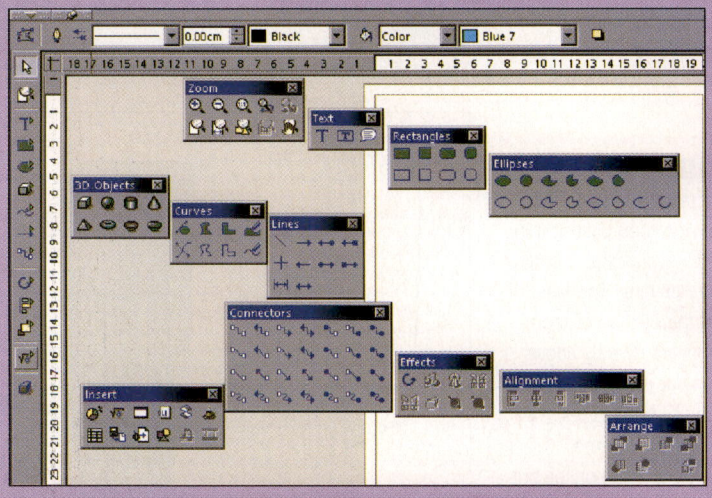
The 'pen nib' icon brings up a menu where you can adjust the properties of any line or shape in your drawing. Note that shapes have a line around the outside. Set the properties to Invisible to get rid of it.

The 'arrow head and tail' icon lets you put various shapes on the beginnings and ends of lines.

The drop-down box to the right sets the type of line. You can have dots, dashes and various combinations. The thickness is set in the next box. You can also type values directly into this.

The colour of your line follows. You have a broad range of ready colours in *OOo*, you have to add extra colours to *OOo* before you can use them. There's a Color dialogue under Area (if you right-click). Add a new colour and it will also be available to colour lines.

The Paint Bucket is the tool to fill the inside of shapes. The drop-down box has your fill options: Color, Gradient, Hatching and Bitmap. Change it to Hatching, then select the hatching type in the next box to the right and it will change. The last icon on the line turns on the dropped shadows. Again, you can set the shadow qualities simply by right-clicking and selecting Area.



The sub-menus. The line of tools that are shown on-screen directly above the working area changes depending on what is selected.

All *OOo* toolbars that have a tiny green triangle can be expanded – there's a sub-menu that appears. If you click on the dark top bar of the menu, it will stay open, which is very handy.

THE ZOOM TOOLBAR

Useful to leave open while working. The "+" magnifying glass sign zooms but you have to click somewhere before it works. The "-" version works instantly. The other icons are specialised Zoom tools, giving you a 1:1 view, previous zoom level and an undo, a tool to display the entire width of a drawing (called a Slide by *OOo*), show objects at their maximum and minimum sizes and a hand tool to move the display to one side or the other.

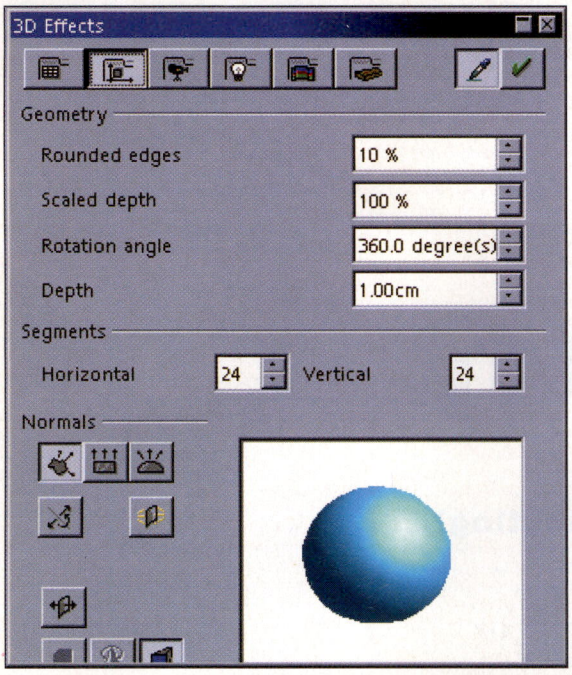
TEXT TOOLS

The "T" gives you a text insertion point. Either drag out a box to write in or just type and the box will expand to fit. The 'T in a box' scales your font to the box size. Drag out the box, type in your text, then click on the arrow (or on any blank part of the screen to de-select the Text tool) and click and drag the box to see the text grow or shrink to fit the box. You can have the box either visible or invisible.

The speech bubble (called a call-out) is nice for labels. Click the item you want to label, drag your line to where you want the label to appear and click again. Next click the 'T' in the text sub-menu to type your description into the box. The label properties (thickness, colour and style) can be adjusted with the Line tools.

SHAPE AND CURVE TOOLS

The Rectangles tool draws either filled or empty squares and rectangles. You can change one to the other easily with the paint bucket tools. There are also rounded shapes. A right-click (or F4) will

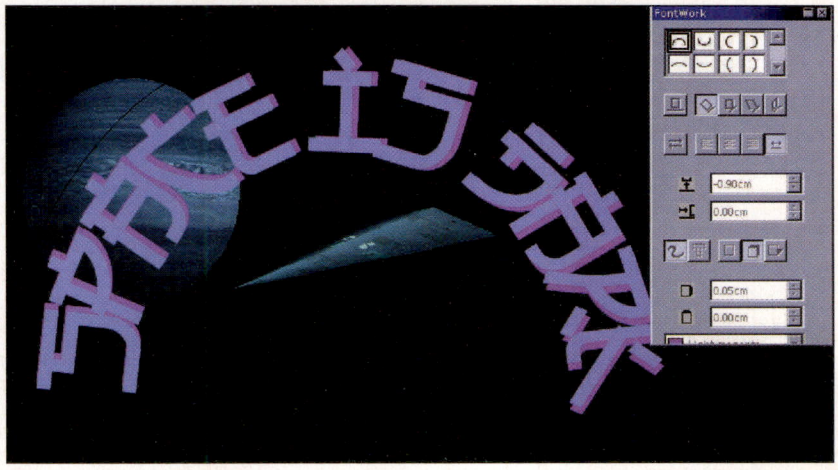


The 3D effects dialogue. This is where you modify the 3D primitives and lighting. Experiment with different effects.

OOo Draw is an amazingly capable program. There are a lot of features that we haven't covered, but there's enough here to get you started. The Help system is very well put together throughout *OOo* – remember to turn on the Extended Tips under Help and briefly hold the cursor over anything you don't understand.

OOo Draw could very easily exist as a separate program, such are its capabilities. If you've even a passing familiarity with *Corel Draw* and *Freehand*, you'll agree that *OOo Draw* isn't in the same class, but it's easy to use and has plenty of features for the average user. It likes plenty of memory too, but that isn't a problem – if a machine is running X, it should be fine.

It's hard to believe that Sun could Open Source such a brilliant tool as *OOo Draw*, but they are obviously convinced that software developed in this way has a myriad of benefits. *OOo* software runs on Linux, Solaris, Windows and a Mac version is currently under trial. Wait a year or so for the word to spread and with luck we'll be looking hard to find any Windows users who still use MS *Office*. *OOo Draw* is a great art resource, part of an excellent suite of office tools. Killer application! ■



Space Is Dark. The starship and planet are 3D while the text is *OOo*'s Fontwork.

allow you to access the Position and Size menu. Here you can set the corner angle and make the shape slant. There's no preview – click OK to see the effect.

The Ellipses Tools draws circles and arcs, either filled or empty. The arcs (or pie-slices) are drawn by dragging out a circle, then placing one point, moving along the edge of the circle to the other and clicking. Again, you can alter the line and fill styles.

3D Objects places a primitive (that's a basic shape, not a monkey) on your drawing. Click and drag your square (or whatever shape you want) and alter it with the shaded 3D square at the bottom of the left side toolbar.

The Curves sub-menu is one that you really need to master. The "circle with a line across its edge" icon makes a closed Bezier curve. Click and drag to your first Control Point, then continue to click to define your shape. Double click to automatically join where you are to the starting point. The crossed line icon underneath the top icon in the sub-menu makes an open line. Double-clicking ends the line where it is. The filled polygon icon makes a closed shape with straight edges and the "L" shaped icon will make a closed shape with 90 degree angles. I used it to draw around the base part of the monitor I made. Finally, there's a filled curve tool that would allow you to draw a shape and it would fill in the last line when you double-clicked.

LINE AND EFFECTS TOOLS

The Line tools are just click-and-drag. The cross is for drawing 45-degree lines. The dimensioning tool is here too. *OOo Draw* comes with 28 types of ready-made connectors to put into your drawings. Click and drag to place one.

The Effects sub-menu is worth keeping open. The Rotate icon places an target point in the centre of a selected shape or line – the centre of rotation. You can move it so that your shape rotates around a point that you define. The cursor changes to either a part circle with an arrow or two fine parallel arrows to show that a 3D rotation is possible, depending on what is selected and where the cursor is. The Flip icon mirrors the original along one axis, which you can move.

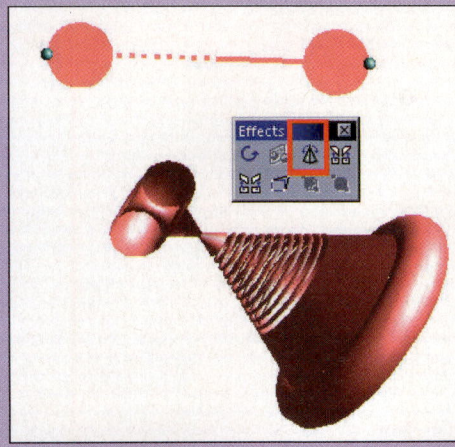
Place your cursor over one of the green boxes that surround the selected object. It then changes shape and you can click and drag the object to the new position.

The icon to make a 2D object 3D can be seen in action in the screenshot shown below, highlighted in red. The other effects tools wrap a shape around a circle, distort a shape and can interactively make and alter gradients.

ALIGNMENT, INSERT AND ARRANGE

The Alignment tools place the selected object either along the top, side or bottom of your page. Keep the Arrange sub-menu open all the time if you are making a complicated drawing. Vector drawing programs make each new shape or line appear in front of a shape or line drawn before it. If you want your latest shape to be partially covered by a shape drawn earlier, send it backwards until it is behind the earlier shape, or send it to the back so that it is behind every shape.

When I drew the space scene, I mapped the stars onto a flat square. It hid everything in the scene, so I sent it to the back. The Insert sub-menu allows you to put photos, graphs, calculations and more into your drawing.



I made a connector into an interesting 3D shape. I also made the keyboard for the computer picture. It's a pity that you can't export 3D objects, although you can import DXFs and a host of other graphics formats.

hw=wh

(hostway is web hosting)

"it's that simple"

KEY FEATURES	DISK SPACE	MONTHLY TRAFFIC	DB/SCRIPT FEATURES	ASP3 SCRIPTING	ACCESS DATABASE DSN(S)	EMAIL FEATURES	POP3/EMAIL ACCOUNTS	SMTP/OUTGOING MAIL	UNLIMITED ALIASES	EMAIL FORWARDING	ADDITIONAL FEATURES	PERL 5 ACTIVE CGI	XML SUPPORT	CGI-BIN and SSI	MEDIA FEATURES	REAL MEDIA PROFFESIONAL	WINDOWS MEDIA SERVER 8	MONTHLY FEE
WINDOWS SILVER	100	1	Yes	0	5	Yes	Yes	Yes		No	Yes	Yes		No	No	£8.95		
WINDOWS GOLD	200	4	Yes	1	10	Yes	Yes	Yes		Yes	Yes	Yes		No	No	£14.95		
WINDOWS GOLD PLUS	200	6	Yes	2	30	Yes	Yes	Yes		Yes	Yes	Yes		No	No	£19.95		
WINDOWS PLATINUM	500	10	Yes	2	50	Yes	Yes	Yes		Yes	Yes	Yes		No	No	£49.95		
WINDOWS PLATINUM PLUS	500	15	Yes	2	100	Yes	Yes	Yes		Yes	Yes	Yes		Yes	Yes	£89.95		

Life has just become simpler here at Hostway! (and we'd like to make your life simpler too)

If you recognise the name IARNA, then you already have a good idea of what we're about as a hosting company, however we'd like to take this opportunity to make something official: our new name in the U.K. is **Hostway**.

As part of the **Hostway** Corporation, award winners and recognised specialists in shared and dedicated web hosting since 1996, we offer a friendly and flexible service to our 100,000+ customers, who are as varied as they are numerous. In the U.K. they include everyone from individuals and developers, to public sector bodies, SME's and well known blue chip clients like Capcom, Chrysler, Mercedes-Benz and Motorola.

When it comes to the business of web hosting though, we think it's the simpler things that matter the most. That's why at **Hostway**, we stand for investment in technology (think usability), quality of service (think dependability) and value for money (think price). We are always working to make web hosting easier to use, more reliable, and affordable to our customers. It's no wonder then that we are becoming synonymous with web hosting itself.

So when you're next looking for the **right** host to fulfil your requirements, the decision might not be as complicated as you think...

Hostway – be it information, application or database driven, nothing is beyond your reach!

web hosting • ecommerce • dedicated servers • resellers **0808 180 1880 • sales@hostway-uk.com • www.hostway-uk.com**

HOSTWAY
THE HOSTING COMPANY

COMMAND LINE

Discover the power of the shell

Worried that you might have to type in commands rather than just point and click? Don't be, says Nick Veitch. Using the shell gives you more control over your PC.

You might be worried about this 'shell' business if you are new to Linux. Typing in commands may seem old-fashioned and complicated, and not the 'modern' way to use a computer, but the truth is that the Linux shell is incredibly powerful and can help you get the most out of Linux.

There are a variety of command shells available for Linux, but the most popular is one called Bash. If you open up a terminal or shell Window in KDE or Gnome, this is most likely the default environment you get. On KDE, the terminal is opened by clicking on the black monitor screen a couple of icons along the toolbar at the bottom of the screen. A window opens up and your shell starts – a brief prompt and a flashing icon now awaits your input! But how do you start, what can you do and what if you do something wrong?

Shell commands

The very first command you should learn how to use is **man**. This is short for manual, and gives you access to a wealth of documentation about all sorts of Linux shell commands. Often, software that you install adds its own 'man pages' to your hard disk, so you can call up information on what the software does and how to control it. This is true of every application you run under Linux – they can all be run from a terminal (although a lot will require an available X display to work as well) and usually they can be controlled better too.

The man pages are your best guide on how to use commands, and you can use it simply by typing:

```
man <command>
```

and pressing return. For example, if you type **man man** information will be displayed about the **man** command itself – what parameters (i.e. words after the command) it expects and the options it supports. This page can be scrolled up and down using the cursor key, and you just need to type **q** to finish.

One important thing to mention here is that Linux commands are case sensitive, so typing MAN won't get you anywhere. All filenames are case sensitive too.

Typing in a load of mixed case filenames can be a pain, but help is at hand. One neat feature of the Linux shell you probably won't have come across elsewhere is completion. Say you wanted to use the **less** command to read a file.

The file you want is called 'incREDIBLY-tedious-long.Filename.txt'. All you have to do is type:

```
less inc<TAB>
```

where **<TAB>** indicates that you press the tab key. You'll find that Linux automatically fills in the rest of the filename for you. If the result is ambiguous (you might have several files that start off the same way but have different endings), the shell fills in as much as it can, then lists the alternatives. This is a really handy, time-saving feature.

Some commands might not have man pages to help you, but almost all have some sort of built-in usage information. The convention for this in Linux is to run the command followed by '--help'. That's two hyphens or minus signs, followed by the word help with no spaces. Try this with **cp**:

```
cp --help
```

and you'll get a brief summary of the options for the copy command. This works for pretty much every common command and piece of software, and at the very least will tell you what sort of options the command is expecting.

Another useful feature of the shell is wildcard support. This is similar to DOS wildcard support. Say you wanted to move a load of pictures from /mnt/camera/ to /home/nick. There are 20 pictures, and they are named picture1.jpg, picture2.jpg and so on. Enter the wildcard. These are special symbols used to match filenames and the simplest is *.

The asterisk symbol means 'any pattern of characters', so to move all files with the .jpg extension, you could type:

```
mv /mnt/camera/*.jpg /home/nick
```

This would move all .jpg files, but to restrict it to our sequence we could fill in some more information, like this:

```
mv /mnt/camera/picture*.jpg /home/nick
```

You can also use the **?** symbol to match any single character or number. There are plenty more wildcard options, but that should keep you happy for a while! ■

Commands you should know how to use

Name	What it does	DOS equivalent	Example
ls	Lists files in current directory	dir	ls -al /usr/
cp	Copies a file	copy	cp abcde.txt /home/nick
mv	Moves a file	move	mv abcde.txt /home/nick
rm	Removes a file	erase	rm cache.tmp
cd	Changes directory	cd	cd /mnt/cdrom
less	Views a text file	more	less README.txt

MOUNTING DRIVES AND DEVICES

Mount up!

In this tutorial Nick Veitch tackles a problem that lots of people encounter – that of mounting drives and other devices on their computers.

Out of all the problems that Linux users seem to have, difficulty with devices seems to be the most common. Mostly it's the actual mounting of devices, and using devices operating with non-Linux filesystems. With this in mind, we've produced this short guide, explaining the use of the mount command, fstab and the filesystem in general.

The first thing that causes some concern among people migrating from other Operating Systems is where the devices are actually mounted. Where are the drive letters and the desktop icons for the drives?

Locating devices

In a Linux system, all devices are mounted somewhere on the root file hierarchy. Let me just repeat that: all of them. Every file is contained somewhere under the hierarchy of root directory, / (not to be confused with /root). In some sense or another, Linux treats all filespace as belonging to one large hierarchy, rather than on scattered elements such as individual drives or shared folders on servers.

If it helps, you can think of the / as the 'My Computer' folder of Windows, though this isn't a completely accurate analogy. Anyway, the different devices are located underneath it.

Options

What the different mounting options mean

async	All file access is done asynchronously.
atime	Update the access time for inodes for every access.
auto	Can be mounted at boot or with the mount -a command.
defaults	Sets rw, suid, dev, exec, auto, nouser and asnyc.
dev	Interprets special character block devices.
exec	Allows execution of binaries and scripts (you normally want this on)
noatime	Do not update inode access times (gives a marginal speed increase for filesystems in heavy use, such as mailservers).
noauto	Will only be mounted with a specific mount command.
noexec	No permission to run executables.
nosuid	Ignores set user identifiers on files.
nouser	Ordinary users aren't permitted to mount/unmount this device.
remount	Remounts the filesystem. In other words, if it is already mounted it will be unmounted first, then remounted with the given options.
ro	Filesystem is mounted read only (for example, a CD).
rw	Filesystem is read/write.
suid	Allows set-user and set-group bits to be used (in other words, for software which runs under a user name different to that of the user who ran it).
sync	All access to the filesystem is synchronous.
user	Allows a normal user to mount the drive (you don't have to be root).

There are many filesystem-specific options, for which you should consult the mount manpages (just type 'man mount' in a console window).

But where are they? It all depends where you, or your distro, has mounted them. Your additional Linux partitions (for example, /home, /usr and so on, if you have them) are mounted seamlessly under the respective directories. And if you haven't already created separate partitions for directories such as /home, Linux will merely append them to the main root filesystem.

For additional drives such as CD-ROMs, non-Linux partitions and removable drives like Zips, most distros will place their mountpoints into a directory called /mnt. This means that the following command:

```
Less /mnt/cdrom/intro.txt
```

is really listing the intro.txt file from your CDROM drive, not the main Linux partition. If it helps at all, you can change the mountpoint for your CD-ROM drive to be /mnt/E or whatever you are used to. And how do we do this?

fstab

When Linux boots up, it checks for a file called /etc/fstab. The fstab file (short for Filesystem Table) lists all the devices to be mounted, where on the root filesystem they are to be attached, and any other special options.

An example line might read:

```
/dev/hda6  /home  ext2 defaults  1 2
```

There are six fields to consider: device, mountpoint, filesystem, options, dump and check.

The six fields

The device is the Linux devicename of the drive and partition to be mounted or, alternatively, the network location of an NFS or Samba networked filespace (more on this later). For local

devices, this will be a pointer to the device in the /dev/ directory on your Linux box.

Briefly, 'hda' is the first IDE device on your machine, followed by hdb, hdc and so on. For SCSI devices, the sequence is similar but the sequence is sda, sdb and so on. The number following the device indicates the partition number so, in our example above, we are looking at partition six on the first IDE drive attached to the machine.

The second field is the mountpoint, or where the drive's file hierarchy is attached to the main root hierarchy. In this case we are using the partition for our /home/ directory, where individual user directories are created, often along with ftp folders and Apache web server directories. We could specify any valid path here, but the relevant directories must already exist on the filesystem. This means if you want to mount something at /mnt/backup, the directory 'backup' must already exist in /mnt before you try to mount it.

The next field tells Linux what sort of filesystem is used on the device being mounted. For native Linux partitions, this is normally ext2 (although some people may now be using reiserfs). Linux can support a huge range of filesystems, including those commonly used by other operating systems such as MacOs (hfs), and Windows (vfat).

For some filesystems, it may be necessary to compile specific support into the kernel. Hfs for one is not commonly included in kernels built by distros.

Mount options

The mount options are used to control how the mount is managed. These options are detailed more fully in the man pages for the mount command, but a brief summary of some common ones is shown in the table here.

Most often the defaults will be used for ordinary devices. CD-ROMs should always be mounted with the option 'ro' to avoid annoying messages. One specific option which is useful if you are going to be accessing Windows partitions is 'umask'. This sets the bitmask of permission bits not present on the Windows filesystem and is analogous to the bash umask command (and the octal mode of chmod). As an example of this, to allow read, write and execute permission for all users, a value of 777 should be used.

The dump field contains either a 1 or 0, and is used to indicate whether the contents of the drive should be backed up by the dump command. As dump isn't used that much, it usually doesn't matter what you set this to, but 0 is safer.

The check field indicates in what order the mounted filesystems are checked when the Linux OS boots up. The root partition should always be 1, and other drives that should be checked should contain a 2. This is usually only worth doing with local partitions (you really don't want to be running fsck on someone else's server) and is obviously irrelevant for things like CD-ROMs. You may as well also set any vfat partitions to 0, as fsck won't check them anyway.

Special cases

Networked drives are really special cases with regard to mounting. These have no corresponding device under Linux, so the device field in the fstab file is actually the network location of the drive.

For example, say there was an NFS mount on a network for the directory /plop. The device specification would be:

Filesystems explained

How to specify your filesystems

Ext	The original Linux filesystem, now superseded. You shouldn't use this any more!
ext2	The default Linux filesystem.
minix	The old teaching OS has been supported for years.
hpfs	OS/2-style 'high performance' filesystem.
msdos	The old msdos-style partitions are supported with this. For Windows partitions, use vfat.
nfs	The network filesystem. A simple and relatively insecure system used by many zUNIX systems for filesharing.
smbfs	The Samba filesystem, which supports shared directories on SMB-based servers.
swap	Linux-specific filesystem, used for its virtual memory operations.
vfat	Windows filesystem used by Win95 and up.
hfs	The partition system used by MacOS.
iso9660	Standard filesystem for CD-ROMS.

```
rich.lxf.net:/plop
```

or:

```
172.18.42.122:/plop
```

if we wanted to use a network IP address. The rest of the fstab entry would be the same as usual, except there are some special options referring to NFS mounts (see the NFS HOWTO for more details on this).

For Samba shares, the format is slightly different. Precede the server name with two forward slashes and append the path of the shared volume:

```
//SMBSERVER/Linux
```

Other special cases are the proc filesystem which is used for process information and /dev/pts, but you can just let the OS take care of these.

Mount and umount

The mount command can be used in three ways. With the -a switch, it tries to automount all devices that are listed in fstab. The second way is to merely give it either the mount point or device as a parameter, in which case it checks fstab for the rest of the information it needs in order to mount the device.

The third general usage for the mount command is probably the most useful for temporarily mounting specific devices. Its format is very similar to the fstab layout:

```
Mount -t vfat -o defaults,user /dev/hda7/
/mnt/windows_d
```

This will mount the filesystem on partition seven of the first IDE device at the mountpoint specified. The type and options are given following the -t and -o switches respectively.

The umount command will, as you've probably already guessed, unmount a mounted filesystem. It is usually used in conjunction with /etc/mtab, in which case you only need to supply the device or mountpoint of the filesystem to unmount it. Removable drives will require you to unmount them before they will eject.

Further reading

This has, we hope, given you some more background on mounting filesystems. If you want to find some more information on Samba and NFS, the best sources are the HOWTOs maintained by the Linux Documentation Project, which can be found at the (www.linuxdoc.org) website.

If you need more specific information for filesystem mount options, see the man pages for mount, which contain a more comprehensive listing. ■

BACKUPS MADE EASY

Backing up partitions

Backing up isn't hard to do with Nick Veitch's guide to partition imaging, so go and save your data!

We have often gone on at length about the necessity of keeping backups as protection against unforeseen events. You don't realise how valuable your data is until you miss it!

Backing up files is one thing, but it is often more useful to back up entire partitions. Restoring the root partition of a drive from a file backup can be tricky, and what do you do if the host system has no viable installed OS?

The solution is to back up entire partition images, although file backups are also useful. There are further advantages to this approach. If you have a set of identical client machines (in a web café environment, for example), you can simply have one backup image which can be used to restore any of the clients. This can also make upgrading clients much easier – install and configure the software correctly on one and you can then use an image to update all the others. With the right software, you could even do this over a network, simultaneously.

Networking
Getting connected

As mentioned, it is possible to set *Partition Image* up to work over a network. Then you can have a central server with your image files, making it much easier to centrally control the data that you back up and restore.

The *partimaged* program runs the server, and all you have to do is open a shell and run it. The user 'partimag' should exist on your server, so you have to create this before you run the server for the first time.

A screen will open showing the status of the 10 allowed connections, and that's it – the server is ready for operation. Remember to make a note of the IP number if the server's hostname isn't included on DNS.

Bear in mind that the limiting factor of backing up over the network is liable to be your network speed. Backing up a 10GB partition over a 10Mb line may take you more than a few hours.

Using the dd tool

For simple partition backups, you might have used the *dd* tool. This versatile tool is used for copying data blocks from one device to another or to a file. For example, to back up your /boot partition which lives on /dev/hda1 you might use:

```
dd if=/dev/hda1 of=$HOME/boot.img
```

This quite simply copies the blocks of /dev/hda1 until it reaches the end. It is possible to specify further options for *dd* of course, but these often aren't useful for copying partitions.

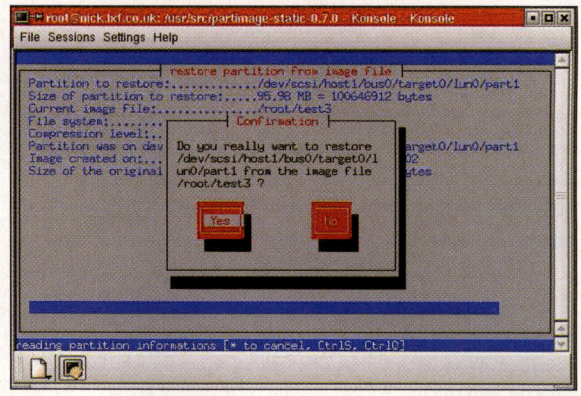
While *dd* is perfectly useful for small images like floppy disks, boot disks and your /boot partition, it isn't quite so useful when it comes to larger partitions. For example, if you want to duplicate a 2GB partition you are going to need another 2GB of filespace somewhere else. This is still the case even if only a small portion of the partition is actually in use. You can retroactively compress the image file, but you still need the disk space to begin with.

Enter Partition Image

These are the kind of limitations that *Partition Image* was developed to overcome. It can dramatically reduce the time and space required to complete partition backups in two ways:

- It only backs up used blocks in the partition. This alone is usually a major space saver. Few people run their partitions to 100% full, so there is space to be saved here.
- The software can also use a variety of compression tools to reduce the size of the image being created.

For all but the very simplest of image files, using *Partition Image* is much more effective, both from a time and storage point of view. The time penalty incurred by compressing the data is often made up for by the fact that you are compressing a smaller file, and the subsequent reduced time taken to copy the image onto backup devices where necessary.


File Sessions Settings Help
restore partition from image file
Partition to restore:.....
Size of partition to restore:.....95.98 MB = 100646912 bytes
Current image file:.....
File system:.....
Compression level:...
Partition was on dev
Image created on:...
Size of the original
Confirmation
Do you really want to restore /dev/scsi/host1/bus0/target0/lun0/part1 from the image file /root/test3 ?
Yes
No
reading partition informations [* to cancel, Ctrl-S, Ctrl-Q]

Restoring a partition should be straightforward, and quicker than the original backup.

Required libraries

libz	*gzip* compression library www.zlib.org
libbz2	*bzip2* compression http://sources.redhat.com/bzip2/
liblzo	fast compression library http://www.oberhumer.com/opensource/lzo/
libnewt	*curses*-like GUI system ftp://ftp.redhat.com/pub/redhat/linux/code/newt/
libslang	*slang*, required for *newt* http://www.s-lang.org
libssl	*OpenSSL* http://www.openssl.org
libmcrypt	*Mcrypt* for secure networking http://mcrypt.hellug.gr

The story so far

Francois Dupoux and Franck Ladurelle released version *0.2* of *Partition Image* in May of 2000. Since then it has grown from its fairly basic beginnings to support more and more filesystems, additional compression formats, and an easy to use GUI, based on the curses-like *newt* system.

You can download *PI* from the homepage, which is found at (www.partimage.org). There are a number of ways to install *PI*, and a number of packages available. If you just want to run the latest stable version, you are probably best off downloading the static binaries.

If you want to compile it yourself either because there isn't a binary available or you want the latest CVS version, there are a few libraries you will need – see the *Required Libraries* box.

To compile from source (once have installed the libraries), you can follow the usual procedure of running **configure** and **make**. However, you may want to configure with options for additional filesystem support, or to control the location of SSL or disable SSL on the server. Running **./configure --help** will give you a full list of the options. So, for example, you might want to say

```
./configure --enable-xfs
make
make install
```

If you have enabled SSL, you'll want to make the certificates:

```
make certificates
```

Partition tables

Before we continue to talk about backing up partitions, we need to discuss how to back up partition tables too.

The partition table is the information stored on the drive which tells the OS where, what type and how many partitions are on the drive. As you'll probably know if you've ever partitioned a disk, there are all sorts of ways to divide one up. Without the information on how a disk is partitioned, it's going to be a lot harder to restore images if you need to.

The *Partition Image* software itself does not (yet) save the partition information, so you'll have to save this yourself.

For a disk which has a few primary partitions, this is not too difficult. The original DOS specification for partitions allowed for up to four partitions on a single device. Under Linux these would be, for example, hda1,hda2,hda3,hda4. These types of partitions are known as primary partitions, and every disk will have at least one. The information describing these partitions is included in the MBR (Master Boot Record) of the device, which is always contained in the first 512 bytes of the disk.

But having only four partitions became a bit of a limitation, especially when much larger disks came along. The solution was to create logical partitions inside one of the primary partitions. This works by allocating a primary partition, then chaining the data for the logical partitions inside them. The first logical partition points to the location of the next and so on. That's why you may often find a disk with partitions labelled hda1,hda5,hda6,hda7, but with no hda2. The numbers 1-4 are reserved for the primary partitions.

Having these linked partitions is a bit of a pain because there is no single part of the disk to read the data from. So in order to save the partition info for the disk, we need to do two things:

1 Back up the Master Boot Record.

2 Back up the information for any logical partitions.

There are one or two utilities to help with this. If you are using Mandrake 8.x, you can use the *diskdrake* utility to store the partition information to a file for you. Select the drive and click on the "More" button in the bottom of the Window. This stores the data in a fairly human readable form too.

The alternative is to back up the data using standard Linux tools. The first task is easily accomplished using *dd*. As we are only talking about a 512-byte file, it's more than up to the task. For example, to save the MBR from your first IDE hard drive:

```
dd if=/dev/hda of=hda.mbr bs=512 count=1
```

To back up logical partitions, you can use *sfdisk*, one of the *util-linux* tools packages which will be installed as part of almost any distro. If you don't have it, you can get *util-linux* from your local kernel mirror or the main www.kernel.org site.

You should be careful with *sfdisk*, as it is possible to lose data using it. Check out the man pages for the options if you are unsure. To back up the partition information, use:

```
sfdisk -d /dev/hda > hda.sfd
```

Compression methods comparison

Find the most suitable compression for your data

type	time taken	size
No compression	142s	81MB
lzo compression	144s	43MB
gzip compression	146s	41MB
bzip2 compression	155s	38MB

These simple tests serve as a guide only. They were performed on a PIII/850MHz machine with standard IDE hard drives. The differences in time taken and final size will differ depending on the type of data stored in the partition and the speed of your processor.

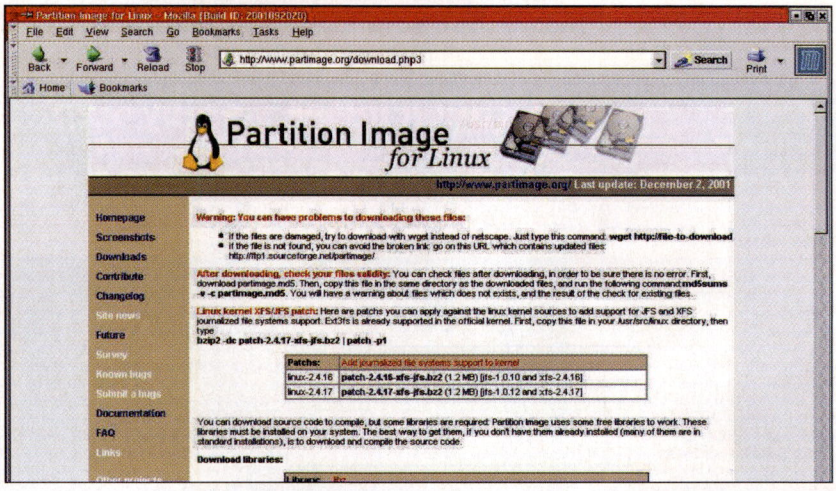
Hard drives

Check your drive configuration for optimum speed

Before performing a disk intensive task like backing up or restoring a partition, it might be a wise idea to make sure that your drives are configured most efficiently. Normally, if you just install a Linux distro and don't check the settings, things will be set to the lowest common denominator. This works, but doesn't get the most out of your hardware.

The first and most basic thing to check is that your system is using an appropriate bus speed. You'll probably find that by default your IDE bus speed is set to 33MHz, which is a safe value for PCI/IDE systems, but you may want to change this. The bus speed setting can dramatically effect the timings for pio modes if enabled on your drives.

You should also check whether your drives are using PIO modes, DMA or whatever. The *hdparm* utility can be used for setting these modes (it's part of the *util-linux* package you can find on the kernel website and its mirrors). However, you can really screw up your disk using this, so read the documentation! If in doubt, best stick to slower backups...



The website contains up to date information, usage guides and many different versions of the software itself.

This will dump the partition info into the file hda.sfd. Armed with these two files, you should be able to reconstruct the partition table of any disk. So for goodness' sake, copy these files somewhere safe. Please don't leave them on your hard disk – they won't be much use to you there if that's the disk or partition that has a problem! Best to keep them on a floppy (or a couple of floppies for extra security), or better still, burn them to the bootable CD you're going to make later...

Backing up a partition

Before we start backing up a partition, you should first know where the partition is. *Partition Image* uses devfs-style partition labels, so if you are backing up hda1 for example, it will be listed as ide/host0/bus0/target0/lun0/part1. For hdb1, the listing would be ide/host0/bus0/target1/lun0/part1. There is a little help in that *Partition Image* will also display the filesystem installed and the partition size. Remember that the partition should really be unmounted before backup, to ensure that no files are open or can change during the backup.

Restoring partition information

If your partition information has been lost, or you want to duplicate the partitions from an identical drive, then we can restore the information we saved earlier:

```
dd if=hda.mbr of=/dev/hda bs=512 count=1
sfdisk -O changelog.sfd /dev/hda <hda.sfd
```

IMPORTANT: This will rewrite your partition table. Make sure that you type the commands correctly and use the correct backup files. Incorrect partition data could result in loss of data.

« The *sfdisk* utility has safeguards to make it hard to write partition data to a mounted drive. You should unmount the drive before you write the partition table. Unfortunate things can happen if you change the partition info when files are being accessed on the disk! If you use the wrong file, or discover a mistake, you can usually recover the partition with the aid of the changelog file we created when we were writing the *sfdisk* data:

```
sfdisk -I changelog.sfd
```

Of course, if you have used *Diskdrake* to back up your partition information, you can also use that to re-write the data.

Whichever method you use, you will normally need a reboot for the changes to be noticed by Linux, so do that before attempting to restore any partitions.

Restoring the partition

Once your partition tables are sound, you can begin restoring the partition. This is an even easier task than backing up because there are less options to worry about. Select the partition you want to restore from the list which is shown (choose carefully) and enter the filename of the file containing the data. If you are restoring over a network, you'll have to select the IP or hostname of the server too.

Press F5 and, if you entered a description for the saved file, this text will now be displayed. You can 'simulate' the write, which at least will check for potential problems with the image file (if it has become corrupt, for example, or if it's the wrong size).

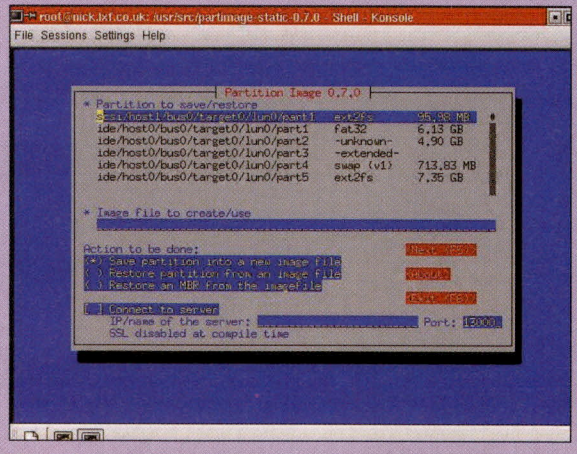
You can also choose to erase empty blocks with zero values. This will overwrite any data in the partition and make it all but impossible to retrieve any of the data from the drive.

Conclusion

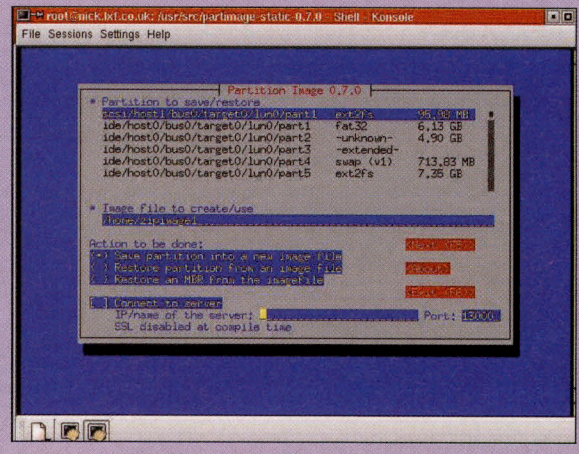
While not exactly fun, *Partition Image* certainly makes the backing up of partitions fairly straightforward and simple. Nobody likes backing up data, but backing up complete partitions offers an easy and convenient solution for those who find it necessary. And after all, if you have all you partitions backed up, what's the worst that can happen? ■

Partition Image step-by-step

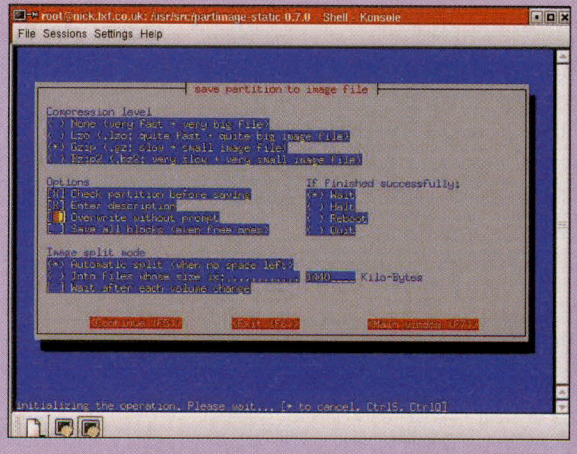
We walk you through the backup procedure


root@nick.lxf.co.uk: /usr/src/partimage-static-0.7.0 - Shell - Konsole
File Sessions Settings Help
Partition Image 0.7.0
* Partition to save/restore
ide/host0/bus0/target0/lun0/part1 ext2fs 95.98 MB
ide/host0/bus0/target0/lun0/part1 fat32 6.13 GB
ide/host0/bus0/target0/lun0/part2 -unknown- 4.90 GB
ide/host0/bus0/target0/lun0/part3 -extended-
ide/host0/bus0/target0/lun0/part4 swap (v1) 713.83 MB
ide/host0/bus0/target0/lun0/part5 ext2fs 7.35 GB
* Image file to create/use
Action to be done:
(*) Save partition into a new image file
( ) Restore partition from an image file
( ) Restore an MBR from the imagefile
[ ] Connect to server
IP/name of the server:
Port: 19000
SSL disabled at compile time

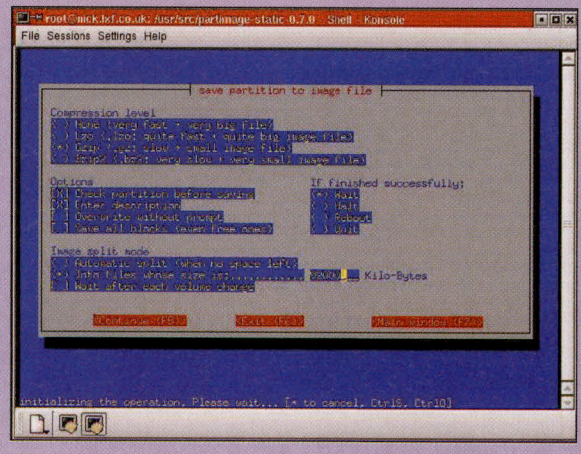
STEP 1 **Run the *Partition Image* program from the console. Make it a reasonably sized window so you can see what's going on! You'll now see a curses-style display of the main menu. The first detected partition will be highlighted. Use the cursor arrows to highlight the partition you wish to back up.**


root@nick.lxf.co.uk: /usr/src/partimage-static-0.7.0 - Shell - Konsole
File Sessions Settings Help
Partition Image 0.7.0
* Partition to save/restore
ide/host0/bus0/target0/lun0/part1 ext2fs 95.98 MB
ide/host0/bus0/target0/lun0/part1 fat32 6.13 GB
ide/host0/bus0/target0/lun0/part2 -unknown- 4.90 GB
ide/host0/bus0/target0/lun0/part3 -extended-
ide/host0/bus0/target0/lun0/part4 swap (v1) 713.83 MB
ide/host0/bus0/target0/lun0/part5 ext2fs 7.35 GB
* Image file to create/use
/home/zipimage1
Action to be done:
(*) Save partition into a new image file
( ) Restore partition from an image file
( ) Restore an MBR from the imagefile
[ ] Connect to server
IP/name of the server:
Port: 19000
SSL disabled at compile time

STEP 2 **Press the TAB key to move between the options. Type in the pathname of the file to store the partition data in. You can leave the rest of the options for the moment, as we are not connecting to a server, nor are we performing any restore options (the default is set to Backup Partitions, as indicated by the asterisk).**


root@nick.lxf.co.uk: /usr/src/partimage-static-0.7.0 - Shell - Konsole
File Sessions Settings Help
save partition to image file
Compression level
( ) None (very fast + very big file)
( ) Lzo (.lzo: quite fast + quite big image file)
(*) Gzip (.gz: slow + small image file)
( ) Bzip2 (.bz2: very slow + very small image file)
Options
[X] Check partition before saving
[X] Enter description
[ ] Overwrite without prompt
[ ] Save all blocks (even free ones)
If finished successfully:
(*) Halt
( ) Wait
( ) Reboot
( ) Quit
Image split mode
(*) Automatic split (when no space left)
( ) Into files whose size is: 1000 Kilo-Bytes
[ ] Wait after each volume change
initializing the operation. Please wait... [* to cancel, Ctrl+S, Ctrl+Q]

STEP 3 **Press F5 to continue: On the next screen we can set the backup options. The first bank of options selects the compression level. The default options are usually fine. If you choose to check the partition (a good idea) you'll soon find out whether you remembered to unmount it first.**


root@nick.lxf.co.uk: /usr/src/partimage-static-0.7.0 - Shell - Konsole
File Sessions Settings Help
save partition to image file
Compression level
( ) None (very fast + very big file)
( ) Lzo (.lzo: quite fast + quite big image file)
(*) Gzip (.gz: slow + small image file)
( ) Bzip2 (.bz2: very slow + very small image file)
Options
[X] Check partition before saving
[X] Enter description
[ ] Overwrite without prompt
[ ] Save all blocks (even free ones)
If finished successfully:
(*) Halt
( ) Wait
( ) Reboot
( ) Quit
Image split mode
( ) Automatic split (when no space left)
(*) Into files whose size is: 2000 Kilo-Bytes
[ ] Wait after each volume change
initializing the operation. Please wait... [* to cancel, Ctrl+S, Ctrl+Q]

STEP 4 **If you want to split the files to fit on CDs or removable disks, you should check these options here. The Wait after each volume change can be useful if you are writing directly to removable media. In the example here we have specified a size suitable for fitting on Zip disks.**

Creating a boot CD for x86

Get to grips with El Torito

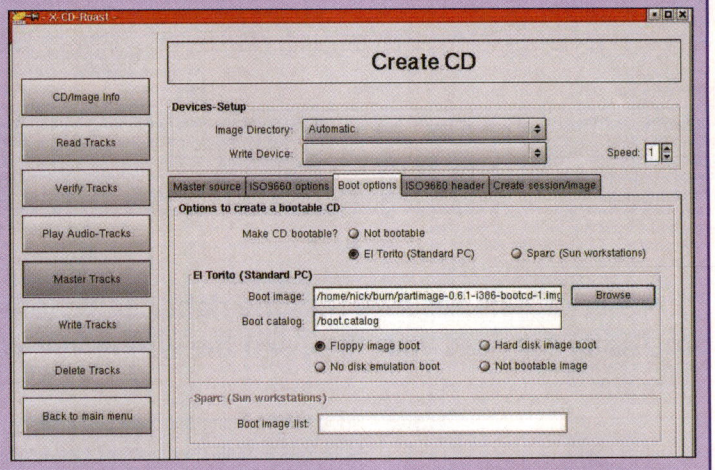
Available from the *Partition Image* website is a useful El Torito image for creating a bootable CD. There are a number of potential uses for this:

- Backing up your computer's root partition (which you won't be able to do when it's mounted)
- Backing up a partition from a non-Linux machine (i.e. a Windows desktop)
- Restoring partitions to a machine with no viable OS

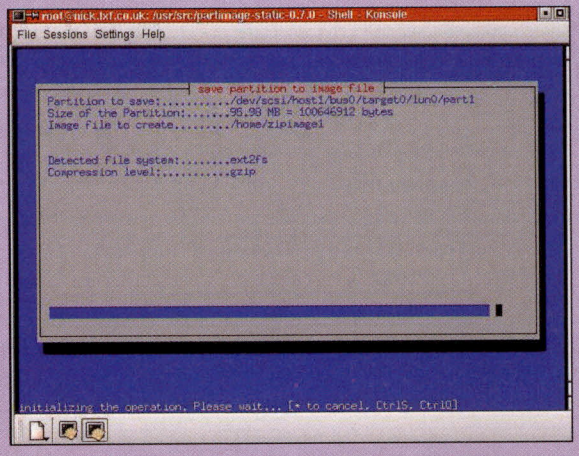
Creation of the boot disk is simple. You just have to copy the El Torito image to a directory and master a CD, using this file as the El Torito image. (For various reasons, this is the name given to the file which is used as the bootable portion of a CDROM.)

If you have a generic partition image, you can burn this to the CD too, enabling you to boot and restore your system without access to any other drives. As the El Torito image takes up a negligible amount of space on the CD, there is 650MB, more or less, of space available for any partition backups.

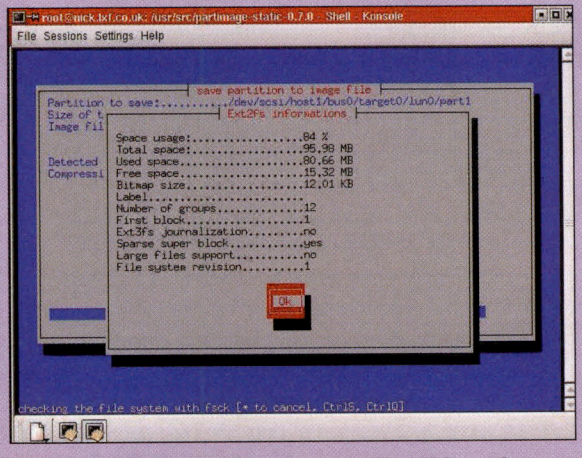
You can create an image and use it with the standard Linux tools, such as *mkisofs* and *cdrecord*, but as GUIs like *XCDRoast* also support El Torito information now, it's probably easier to use this instead.



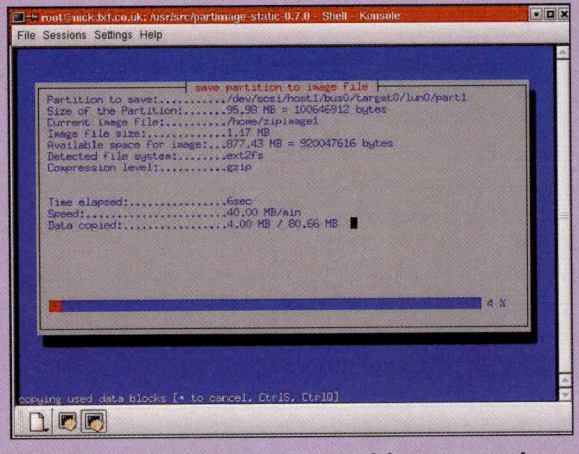
Point *XCDroast* at the El Torito image to create a bootable CD.


root@nick.lxf.co.uk: /usr/src/partimage-static-0.7.0 - Shell - Konsole
File Sessions Settings Help
save partition to image file
Partition to save:...../dev/scsi/host1/bus0/target0/lun0/part1
Size of the Partition:...95.98 MB = 100646912 bytes
Image file to create...../home/zipimage1
Detected file system:.....ext2fs
Compression level:.....gzip
initializing the operation. Please wait... [* to cancel, Ctrl$, Ctrl0]

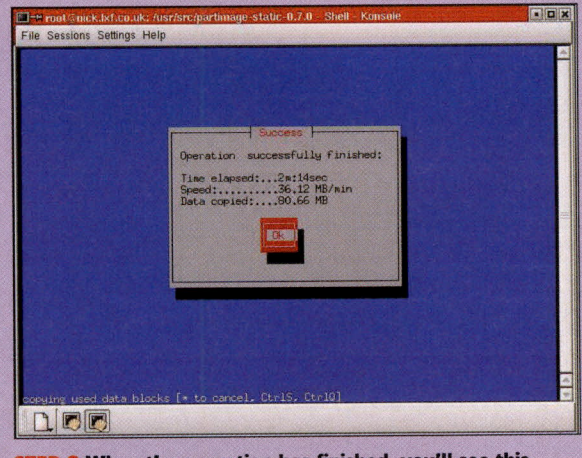
STEP 5 **Hit F5 and you will have the option to enter a useful description. Do so and then the operation will initialise. This is the point at which the filesystem is checked (if you chose that option) so it can take a few moments. All you have to do is wait patiently.**



STEP 6 **An info screen will pop up to tell you about the partition you are backing up. This may have various pieces of data, depending on the filesystem in use on that partition, in addition to some info about blocks and the size and used space on the partition. Press ENTER to continue.**


root@nick.lxf.co.uk: /usr/src/partimage-static-0.7.0 - Shell - Konsole
File Sessions Settings Help
save partition to image file
Partition to save:...../dev/scsi/host1/bus0/target0/lun0/part1
Size of the Partition:...95.98 MB = 100646912 bytes
Current image file:...../home/zipimage1
Image file size:.....1.17 MB
Available space for image:...877.43 MB = 920047616 bytes
Detected file system:.....ext2fs
Compression level:.....gzip
Time elapsed:.....6sec
Speed:.....40.00 MB/min
Data copied:.....4.00 MB / 80.66 MB
4 %
copying used data blocks [* to cancel, Ctrl$, Ctrl0]

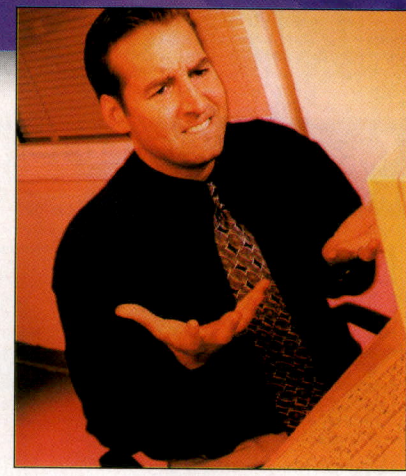
STEP 7 **The operation will now start, giving you a running total of the amount of data stored so far, and various other statistics to keep you happy. The time taken for saving the image obviously depends a great deal on how much data you are backing up!**



STEP 8 **When the operation has finished, you'll see this screen showing you what exactly has happened (note that if you have chosen to wait for split files, you'll get messages in between for media changing). Click on OK and that's it! Your data is backed up.**

BASIC USER SECURITY

Mastering file permissions



Maurice Kelly makes sure that the right people (and processes) can read your files, and the wrong 'uns can't.

Pardon me for being patronising, but did you know that Linux was a multi-user system? Not a multi-user system in the way that the home versions of Windows are (like Win95/98), but one where each user can maintain proper protection of their files. If you're an experimental sort of person, then you may have come across error messages like:

```
bash: sample.txt: Permission denied
```

when you try to create files where you aren't allowed to, or tried to read a file you haven't been given permission to read. This might seem like a strange concept – after all, it's your computer so why shouldn't you be allowed to do what you like with it? Well, it's all for your own good – really!

Like I said, Linux is a multi-user system. If you share your PC with other people then you put your system and your files in their hands, and while they may be the most well-meaning people in the world, an accidental delete of your /etc directory is going to cause serious headaches. You may be forced to share a PC with people who are not so pleasant (in a school, workplace or on the net) where you could be at risk from malicious users.

Permissions are a vital part of Linux life. They work by dividing the world according to a file into three areas – the owner of the file (owner), the group the file belongs to (group) and the rest of the world (others). This model works reasonably well and is quite intuitive. Each area has its own set of permission bits for read, write and execute (**r**, **w**, **x**) which provides nine different bits in total that can be used to control access to the file. To see some permissions on your files, pick a file and type the following at the command prompt:

```
$ ls -l sample.txt
```

The result may come back as shown in the "Who sees what" box below. Each permission flag is explained.

Using permissions we can control access to files to suit our purposes. Say we wanted to allow read access to the file "sample.txt" so that everyone on the system could read it. However, we only want ourselves and the members of the groups "users" to be able to execute the file and only we ("mkelly") should be able to write to the file. Basically we want the permissions to read **-rwxr-xr--** (owner – read/write/execute; group – read/execute; and others – read). We can set the levels using the *chmod* command.

In order to change the permissions we enter the command as follows – **chmod <permissions> <filename(s)>** – substituting the appropriate permissions and filename(s). Permissions can be specified in a number of ways. It is possible to absolutely specify a complete set, or certain types can be independently updated. We want to change the permissions absolutely to those specified above. This is done using an octal (base 8) representation for the permissions. A three digit octal number will represent three categories of users, where each digit is weighted 4 (read), 2 (write) and 1 (execute). So if we want **rwx** – the octal digit is 4+2+1 = 7. Using this principle we need the 754 to achieve **-rwxr-xr--**. Thus the command to give us these permissions on "sample.txt" is as follows:

```
$ chmod 754 sample.txt
```

We can check the result by issuing:

```
$ ls -l sample.txt
-rwxr-xr--  1 mkelly   users        22 Dec  6 22:18 sample.txt
```

As you can see the permissions have been altered successfully.

If you're sharing a secure machine with no hostile users you might want to allow read access to your entire home directory. This could be done by issuing **chmod -R 754** * (where **-R** means to recurse the directory structure) but is a bad idea as you would make every file have permission **754**. You've achieved your desire to make every file readable, but you've also enabled every file to be writable by people in the group "users", which on a standard Linux installation is all other users. Say "sample2.txt" is currently read/write for the owner and had no permissions for everyone else.

```
$ ls -l sample2.txt
-rw-------  1 mkelly   users        22 Dec  6 22:20 sample2.txt
```

If you only want to update some of the permissions – say, read access to everyone – you need to issue the *chmod* command:

```
$ chmod +r sample2.txt
$ ls -l sample2.txt
```

Who sees what

Permissions in a nutshell

```
-rw-rw-rw-  1 mkelly          users       22 Dec  6 22:17        sample.txt
||||||||||    |                |           |                       |
||||||||||    +-> Owner        +-> Group   +-> Modification date   +-> the filename itself
||||||||||
||||||||||
|||||||  +----> read/write/execute permissions for others
|||||||
|||||||
||||  +-------> read/write/execute permissions for the group (in this case "users")
||||
||||
|  +----------> read/write/execute permissions for the owner (in this case "mkelly")
|
+-------------> the type of file. This is a normal file – it could be a link (l), directory (d), or device (c)
```

```
-rw-r--r--  1 mkelly  users      22 Dec  6 22:21 sample2.txt
```

This sets read access for all users, but leaves all other permissions alone. It is also possible to apply an individual permission in this way. Say you wanted to grant execute permission on all your files for the owner, but didn't want to grant to anyone else. This is executed as follows:

```
$ chmod u+x sample2.txt
$ ls -l sample2.txt
-rwxr--r--  1 mkelly  users      22 Dec  6 22:21 sample2.txt
```

So as you can see, it is possible to allocate permissions absolutely using an octal string, or to simply update individual permissions. There is more to the *chmod* command than we can cover here, but have a look at **man (1) chmod** for more info.

The Root User

So that's mortals covered but what about the all-powerful super-user – how does root fit into this scheme? Put simply, root has access to every file on the system, which is why it is so important to log in as a normal user. Root can delete/mangle any files regardless of permissions, and using root as your day to day user account simply increases the possibility of you doing something foolish. File permissions apply to files belonging to root in the same way as for normal users. If root decrees that you can read/write/execute a file belonging to him/her, either by allowing access to a group or to "others", then you can alter or use that file in the same way that you can access one of your fellow users' files with the same permissions.

While you can override any permissions with root, you have to remember not to go insane. Linux may seem like it has a very strange directory structure, but it is one that has evolved over time. Some file locations are very important, so it's not wise to go moving stuff to new homes at random. Likewise, while it may be tempting to save everything in the root directory, just remember how much easier it would be to delete everything on your system with a slip of the *rm* command while in /. While not every distribution places files in the same place, it is worth a read of the Filesystem Hierarchy Standard at **www.pathname.com/fhs/**.

File Ownership

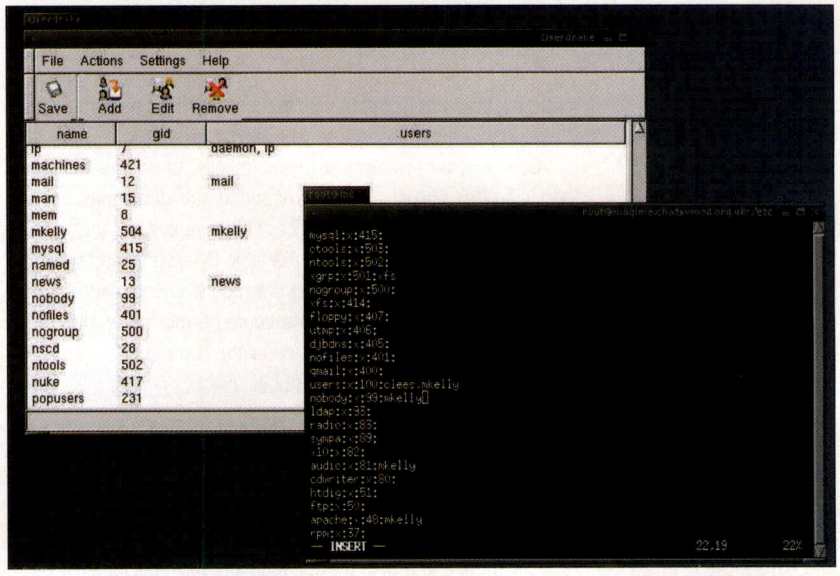
While discussing the permissions above, the concept of file ownership was introduced. All files on your system will belong to someone and many of them will belong to root. When you create a file it will automatically belong to you, and usually your default group. File ownership does not normally change that much as users are tightly restricted in what they can do. Much ownership changing will be done by the root user as he/she can decide the ownership of any file on the system.

The lack of freedom when it comes to changing ownership is not without good reason. If you could change file ownership to any other user it's possible that you could exploit otherwise safe situations creating a security risk on your system. Fortunately, more use can be made of the group attribute, which is more lenient and allows you to assign a file that you own to any group you are a member of. We will use the *chgrp* command to assign our files to other groups in the example below.

Detailed information on both of these commands is available in the man pages **man 1 chown** and **man 1 chgrp**.

Protecting Your Web Docs

Say your Linux box is a web server on a local network or even on the Internet. Your own personal html directory contains sensitive information that should only be accessed by certain people through password-protected pages. Maybe it's not that tight, but you've got a bunch of scripts that you don't want others to see the code behind. You can prevent prying eyes viewing from the web, but what about local users? Proper use of permissions on your own html directory will protect your pages against marauding local users. Of course you need to be careful about how the permissions are set up. You don't want to deny access to the person who maintains the info, but you certainly don't want to be so restrictive as to prevent the *httpd* daemon from being able to read the files in order to serve them up to a web browser.



You can add yourself to more groups by editing the /etc/group file, or using a graphical tool supplied by your distro. The choice is yours.

On a standard Linux box running *Apache*, the personal html directory is called "public_html" and is located in your home directory. When *Apache* receives a request for **http://localhost/~username/** it fetches from the directory /home/username/public_html (or ~/public_html for short). This may not be set up on your system in which case you'll need to do it yourself, or get your administrator to do so.

My *Apache* httpd daemon runs under the user "nobody" and group "nobody". In the remainder of this example where you see ~/public_html, you should substitute your own personal html

»

Default Permissions

Behind the mask

This talk of permissions begs the question, "Do I have to continually use *chmod* every time I create a new file?" On most systems, non-executable files are created with 644 permissions. This, of course, gives read access to all your files to everyone. If you want to avoid this, say by making new files appear as 640, you need to use the shell built-in command *umask*. This is generally set system-wide in /etc/profile (for *Bash* users) but you can override it in your own ~/.bash_profile.

It works by specifying a mask for the creation of new files – you subtract the mask from the value 666 in order to get the default permission of new files. For example, in order to get a permission of 640, you need to subtract 640 from 666, which results in 026. Okay, it's actually not quite that simple – that covers the creation of ordinary files. When an executable gets created it is going to be made executable by everyone by default, so you should add 1 to the mask for every category of users you don't wish to be able to execute the file. So if you had a script or program that you wanted people in your group to be able to read but not execute, and deny everyone else access completely, you will need to add a 1 to the group permission mask and the other users permission mask, making your complete umask 037.

You can set your own mask (either temporarily at the command line or permanently in your start-up file) by issuing umask mask_value (where mask_value is something like 037.) You can view the current setting by issuing umask at the prompt, or umask -S for a more explanatory notation.

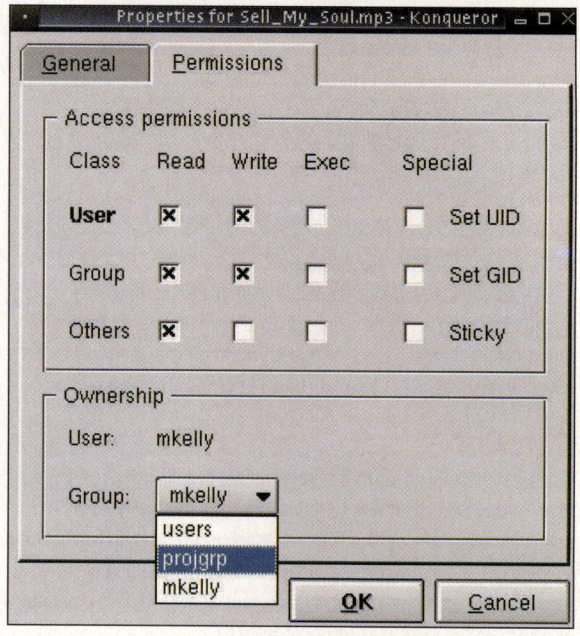
directory, and where you see the username "nobody" and groupname "nobody" you should use your own equivalents. You can check what user/group your copy of *Apache* runs under by checking your httpd.conf file. Where you see the username "mkelly" you should substitute your own username.

On our system we want the user "mkelly" to be able to read and write files within ~/public_html and its sub-directories, but all other users should be denied access (both read/write and execute). The files are currently readable by anyone, and so the first thing we want to do is clamp down on who can read/write those files. To turn off read/write/execute permission on the ~/public_html directory we must enter the command:

```
$ chmod o-rwx /home/mkelly/public_html
$ ls -l
total 1
drwxrwx---  18 mkelly  users  1024 Apr  9 13:25 public_html
```

Perform an **ls -l** as shown above, and note the permissions only extend to the owner and the group. Change into ~/public_html and repeat there and in any sub-directories in order to protect the files below. Now only user "mkelly" and anyone in the group "users" can read or write to the web documents. Unfortunately, while this is a step to what we want, it's partially a step backwards. If you fire up your web browser and point it to **http://localhost/~mkelly/** (substituting your username for "mkelly") you're likely to get a **403 Forbidden** error. To top it all off, most of the other users are still able to read our files in the actual directory as they are members of the group "users".

Fortunately, fixing both these problems can be done with a single command. The *chgrp* command can be used to specify which group a file belongs to. First, though, we need to add our user to the group "nobody". As root, open the file "/etc/group"



Right: ***Konqueror*** **is the file manager for KDE2 and offers the user an easy way to manage permissions for a file.**

Changing Permissions In KDE/GNOME

The way of the GUI

Permissions don't have to be modified from the command line – you can use your GUI as well. In GNOME, start up your preferred file manager – I use *Nautilus*. Navigate to the file you wish to change permissions on, and right-click on it. From the menu choose the "Show Properties" option. A new window will open displaying the file properties – click the third tab (marked "Permissions") and you'll be presented with something similar to what you see in the screenshot above.

You should see a graphical representation of the current permissions on the file – clicking on the tick boxes will cause that permission to be set, and *vice versa*. (The options Set UID, Set GID and Sticky will be explained in the section "Special Permissions".) The "Text View:" and "Number View:" lines should update as you change options and observing this is a good way to familiarise yourself with common modes.

KDE users should start up *Konqueror* and navigate to their chosen file. As with *Nautilus*, you right-click on the file and select the "Properties..." option. The tab marked "Permissions" displays an almost identical arrangement to that produced by *Nautilus*, without the number and text representations of the permissions. As before setting a permission is as simple as clicking on one of the buttons. See the screenshot for a view of the dialog.

The permissions tabs in both dialog boxes also showed the Ownership section. Note that, as described in the section "File Ownership", you can only change group ownership to a group of which you are a member. Until you are a member of multiple groups you won't be able to change the group to anything else.

and find the line for the group "nobody". Add the desired username to the end of the line (see the screenshot above for an example) and save the file. Now change the group of ~/public_html using the *chgrp* command (or your favourite file manager software).

Voila! If we repeat that in sub-directories of ~/public_html, the *httpd* daemon should have permission to read those documents, and other normal users should not be able to read, write or execute those files. Attempting to access those confidential documents as another user should result in something similar to:

```
$ cd /home/mkelly/public_html
bash: /home/mkelly/public_html: Permission denied
```

This was quite useful but there is something you should watch out for. When we create new files we need to set the proper permissions and ownership afterwards. On most systems creating a new file will cause it to be owned by the user, in this case "mkelly", and belong to the group "users". Most systems also create files with the permissions **rw-r--r--.** For example, if I create the file "test.html" in ~/public_html, I can see that it gets created as follows:

```
$ ls -l test.html
-rw-r--r--  1 mkelly  users      81 May  4 22:28 test.html
```

We need to do two things here. We want to get rid of any permissions for others and make it readable to the *httpd* daemon, so we do:

```
$ chmod o-rxw test.html
$ chgrp nobody test.html
```

See the box *Default Permissions* for details on how to generate new files that don't have to be altered using *chmod*.

Group File Sharing

Another use for permissions is file sharing for a group of people. An example situation is that of a shared project where a set of common files are used by the project members. Granting read/write permission to "others" would allow everyone to use the files, but offers no security. A better idea is to create a group (for example, "projgrp") and assign all the files to that group.

Creating a group should be done by root. From the command prompt issue the *groupadd* command with the name of the new group. For example:

```
# groupadd projgrp
```

Your new group will be created. To add each of your prospective group members, we could edit the file "/etc/group" as we did before, but we will use command "usermod" as follows:

```
# usermod -G [existing group1,existing group2,...,]projgrp username
```

Take care in issuing this command – you must specify the user's existing groups otherwise he/she will be removed from them. For example, if the user was already in the groups "ftpuser" and "pppuser", you would issue **usermod -G ftpuser,pppuser, projgrp username**. You can find out what groups a user is already in by using the *id* command at the prompt.

Use the *chgrp* command to change the group of all the files to your new project group. Each member of the group who wishes to contribute files must do this. Now if you restrict file access to user and group only (**chmod o-rwx filenames**) you will find that only the members of your project group can actually edit, view or execute your files.

While groups may seem useless on a single-user system bear in mind that many processes and daemons on your system run as their own user and group (like the *httpd* daemon we saw above), and so assigning files to groups is an added layer of protection for those files without them belonging to root.

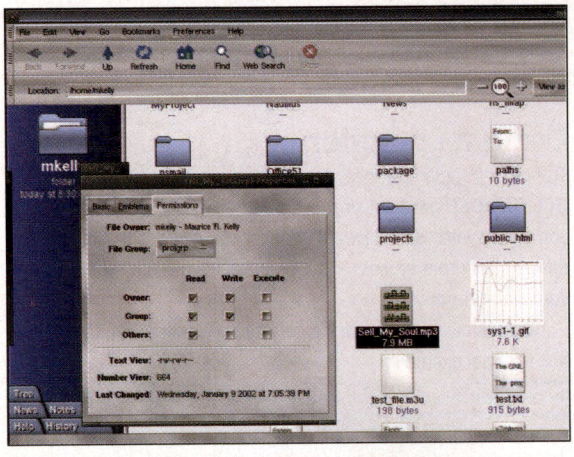
Special Permissions

We didn't quite give you the full story when we said that the permissions were read, write and execute. There are in fact three other special attributes used in relation to executable files – *setuid, setgid* and the *sticky* bit. The *setuid* bit will cause the process executed by the file to run with the owner ID of the file (useful for users who need to use a program that needs root access to a device – for example, *pppd* for modem dialup use.)

Setgid is similar to *setuid* but sets the process to execute with the group ID the same as that of the owner. The *sticky* bit (save text bit) will save the image of the program in the swap memory so that it can be executed more quickly in future – it does not need to be reloaded into swap.

Directories

Directories have permissions too, but are not quite the same as files. Directories use the read/write/execute bits as well as the *setgid* and *sticky* bits (but not *setuid*). A directory with read access will allow the listing of directory contents. A directory with write access can have its files added, removed or altered. A directory with execute permission can have its file details listed (for example, in **ls -l**). Therefore, having read but not execute permission enables you to be able to look at directory contents, but not see full file details. Note that execute but not read permission does not allow any listing of files or file details.



Nautilus is the GNOME equivalent of *Konqueror* and works in much the same way. Permission editing in a file manager is pretty similar no matter what you use.

Using *setgid* on a directory will cause files created in a group to take on the same group ID as the directory instead of the group ID of the file creator. This is obviously useful when you're creating files in your public_html or project group directories in the previous examples. Using the *sticky* bit makes a directory become "append only" – users cannot remove files unless they own them.

In the example above regarding maintaining security of web documents, you may remember that we denied read/write access to all web documents to any users except those we specified as our own user and the members of the group "nobody." However, it is still possible for a local user to look at the directory contents, so we may want to protect those contents from onlookers (at times filenames can give away as much information as file contents). To protect your web documents, you can set the permission **rwxrwx---** on your ~/public_html directory – that way only the owner and members of the group "nobody" can read or write to the directory and this also applies to sub-directories. Now your web documents are safe from both tampering and viewing by unauthorised local users.

Finally...

Permissions can be a pain at times, slowing you down when you just want to get the job done quickly. But the next time you do an inadvertent **rm** and a file permission saves your documents, you may just be glad you protected those files after all! ■

Usedir

Setting up your ~/public_html on Apache

For this exercise you want to be able to use the UserDir feature of *Apache*. To do this, as root, open up your *Apache* configuration file called "httpd.conf" (or in "srm.conf" on older systems.) Search down the file until you see a configuration directive called UserDir. It should read

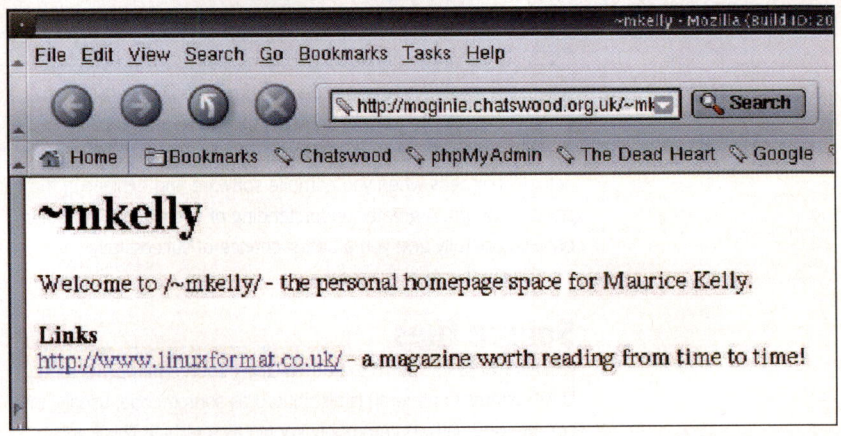
```
UserDir public_html
```

(you may simply have to uncomment it.)

This enables the UserDir feature, but you need to tell *Apache* that it's okay to read documents from there. Below the UserDir directive enter the following text

```
<Directory /home/*/public_html>
    Options Indexes
</Directory>
```

This should be sufficient to allow you to follow the tutorial. Now restart *Apache* to allow the changes to take effect, and try your new html directory out by loading the site http://localhost/~username/ into your preferred web browser.



The result of the ~/public_html directory being accessible to user "nobody" – we're now officially web published!

SOFTWARE INSTALLATION

Compiling software from source

Nick Veitch presents a beginner's guide to the processes and potential pitfalls of compiling software.

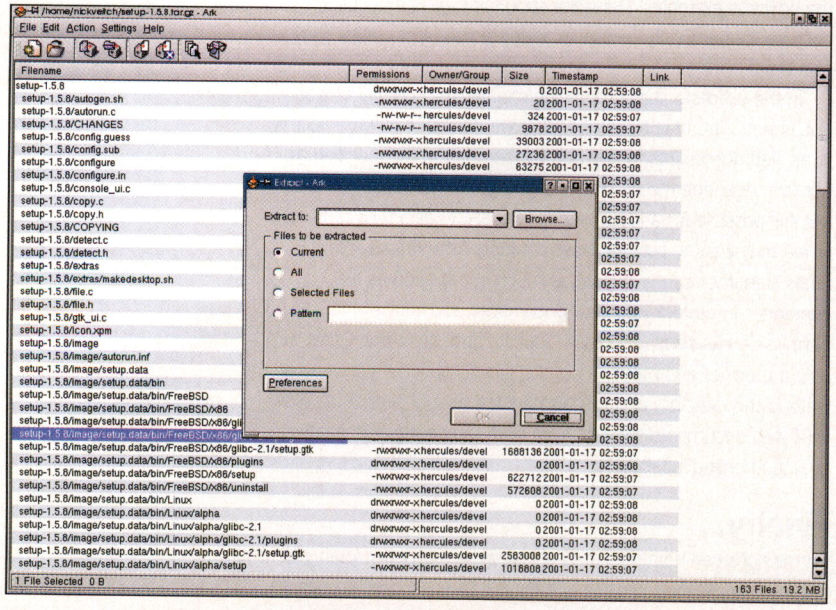
One of the perennial problems that seems to crop up again and again is people having difficulty installing software, usually software they have to compile themselves. In this tutorial we'll be looking at what actually happens when you compile software and explaining the steps as we go. A greater understanding of what is actually going on will hopefully give you a better chance of successfully installing software in the future.

Source files

Most software for Linux and other Open Source platforms (and Unix systems in general) is distributed as source code, usually in a compressed archive commonly known as a tarball. This is just a collection of source code and data compressed into a single file for easy transportation.

As well as being put into the tar archive format, these files are often compressed with *gzip* or *bzip2* to save space and download times. The files will then commonly end in '.tgz' or '.tar.gz' for *gzipped* files and '.bz2' for *bzipped* ones.

Before installing any software, you will normally want to log in as the root user. Only root can install the software globally on the system so every user can access it, so before you go any further, it may be a good idea to open a terminal and use:



If you find it easier, you might consider using a graphical program like *Ark* to extract files from a tarball archive.

```
su -
```

and enter the root password. Now you can copy the tarball file from CD or download it from a website onto your hard drive. A good place to put files like this is in /usr/src directory. Let's assume you want to compile *Gamazons* from a *Linux Format* coverdisc.

```
cp /mnt/cdrom/Games/Gamazons/gamazons-0.77.tar.gz  /usr/src
```

Now we can uncompress it using

```
tar xvzf gamazons-0.77.tgz
```

A lot of text will appear on the screen, showing the files as they are created. Now enter the gamazons directory and you are ready to begin compiling the code.

Compiling

What does compiling software actually do and why does it seem so problematic? A computer program is usually written in a 'language', like C. The computer itself doesn't understand this language. The CPU only understands a limited number of instructions, most of which are simply about moving data from one place to another, or comparing two values. What's more, this instruction set is usually unique to a particular processor. There are slight differences, for example, in the instructions understood by a standard 486 processor and a Pentium, and vast differences between an AMD Athlon and a PowerPC G3 processor, for example. Writing in a language like C, which is then translated into the processor's code, means that the same software can be transported between different processors easily. When running on the same OS, well written code will compile no matter what the processor actually is.

That's the main reason why a great deal of software is available as source code for Linux, because Linux itself can run on so many different processors. If you compiled code for a PowerPC chip and distributed it, nobody with a Zaurus PDA or an Itanium server would be able to run it! Distributing the source means that it can be compiled on many different system. It can also be made to work on other Linux systems with little alteration.

Using a compiler

The most popular compiler for use with Linux is the excellent *GCC*. The *GCC* program simply takes the source code and compiles it into assembly language for the processor (and while this isn't a simple process, it's a problem for the authors of the compile and not you!).

Right. Now we'll consider a very straightforward program, like this minimal example:

```
int main(){
        printf("Hello, readers\n");
}
```

It's a very short C program which outputs a line of text when run and is probably about the simplest C program you could

write. In order to compile it, all we have to do is save this text in a file and then run *GCC*, supplying the filename as an argument:

```
gcc test.c
```

(Assuming we saved the text in a file called test.c.) The program will compile and you will now have a binary, which by default is called *a.out*. If you want to give the output a different name, then you can specify this with the **-o** flag:

```
gcc test.c -o test1
```

Now you have a binary file which you can execute on your computer and it will work perfectly well. However, if you take this binary and mail it to a friend, you may find that it doesn't work for them at all. If they are using a different processor, the code simply won't be understood, and this is why source distributions are often used.

Libraries

Very few pieces of software are like our example above. One of the great benefits of free software is that it is for sharing, and programmers share their code prolifically. The benefit to programmers is the re-usability of code. There is no point spending ages writing code that someone else has already written. For this reason many operating systems rely on libraries of code. You may have seen shared libraries on other platforms (for example, in Windows, all those .DLL files are libraries).

Libraries make code available through functions which the main program can call, just as if the library was included in the program. The example shown in Listing 1 (*right*) uses the *SDL* library (a graphics and sound library used extensively by games programmers) to create a window on the screen.

You can see that the listing is only 30 lines long, but it opens a window on the screen, with a titlebar and gadgets for minimising and closing it. Clearly, the code to do all that is not contained in the listing. It is actually included in the library, along with all sorts of other functions.

In order to make the functions available, the programmer just has to include the header file at the top of the code (the **#include <SDL/SDL.h>** line). But the actual core code of the library isn't included in the program – it is loaded from the library each time you run it. In order for this to work, the compiled program code has to 'link' to the library code.

This means the compiler has two jobs to do – first it compiles object code from the source, then it links it and creates executable code:

```
gcc -c  sdlcode.c -o sdlcode.o
gcc sdlcode.o -L/usr/lib/ -lSDL -o sdltest
```

The first line includes the options to compile the object code. The second line then links this code with the *SDL* library. If you don't specify the libraries to link to, you will get errors as the compiler won't be able to find the code for the functions you have used. The **-L** flag tells the compiler where to look for the library, though most libraries are included in the default library path and this can often be omitted. It is possible to combine these stages into one command, although the compiler actually completes the same two-stage process:

```
gcc sdlcode.c -o sdltest -L/usr/lib -lSDL
```

will result in the same executable file.

The library system works well. It allows programmers to re-use code and it makes the actual source files smaller and more manageable. Imagine a simple test like the one in the listing which did have to include all the code for talking to the X server, opening the window and everything else!

Listing 1
A short SDL demonstration

```
#include <SDL/SDL.h>

main(int argc, char *argv[])
{
  int done=0;
  SDL_Surface *screen;

  if ( SDL_Init(SDL_INIT_AUDIO|SDL_INIT_VIDEO) < 0 ) {
       fprintf(stderr, "Unable to init SDL: %s\n",
SDL_GetError());
       exit(1);
    }

screen=SDL_SetVideoMode(640,480,8,SDL_HWSURFACE|S
DL_DOUBLEBUF);
  if ( screen == NULL )
  {
    printf("Unable to set 640x480 video: %s\n",
SDL_GetError());
    exit(1);
  }
  SDL_WM_SetCaption("This is our window",
"TESTWINDOW");


  while(done == 0){
            SDL_Event event;
            while ( SDL_PollEvent(&event) )
            {
                        if ( event.type == SDL_QUIT ) { done
= 1; }
                        break;
       }
  }
  SDL_Quit();
  return 1;
}
```

However, the downside to this is that if you don't have the library, you won't be able to compile the software. Libraries are the most common dependencies for software, which is why, if you take a look in your /lib and /usr/lib directories, you'll find an awful lot of them.

Library versions are also important. For the most part, libraries are backward compatible. That means that source written for a previous version of the library will usually work on the latest version too. The reverse is not the case though.

For programmers, it is often desirable to use the latest versions of libraries, usually because useful new features have been added or irritating bugs have been removed. This may mean that although you have the right library to compile the software, it may not be the right version.

For the people who actually compile the code, this can be a bit of a nightmare. It can also cause considerable problems for the programmer, not the least of which is because if they are creating large projects using many libraries, the compile

»

command soon gets very long and complicated. Fortunately there is a helping hand for everyone which comes in the form of *Autoconf* and *Automake*.

Configure

These tools can be used to build a configure script for a software source distribution. This script can test for the presence of particular features and libraries, accept particular options which can be used to control features of the software, and create a Makefile which can be used to build the executable.

You'll have no doubt come across the configure script before when trying to compile software. The main functions of this script are to test what sort of system it is being run on, whether the compiler supports various features and what libraries are installed. Usually, all that is required is to run the configure script:

```
./configure
```

Nine times out of ten this will work, and when it doesn't it's usually quite obvious where the problem lies from the resulting error message. It is nevertheless important to read any documentation that came with the source code, particularly software which is intended to be used on many different platforms (even ones which aren't Linux) as there may be vital options, environmental variables or other things that must be set.

Try this is your gamazons directory that we uncompressed earlier. This game uses common GNOME libraries, so if you get any errors regarding these, you may want to update your GNOME packages for your distribution. Assuming this went correctly, all you have to do now is make the file:

```
make
```

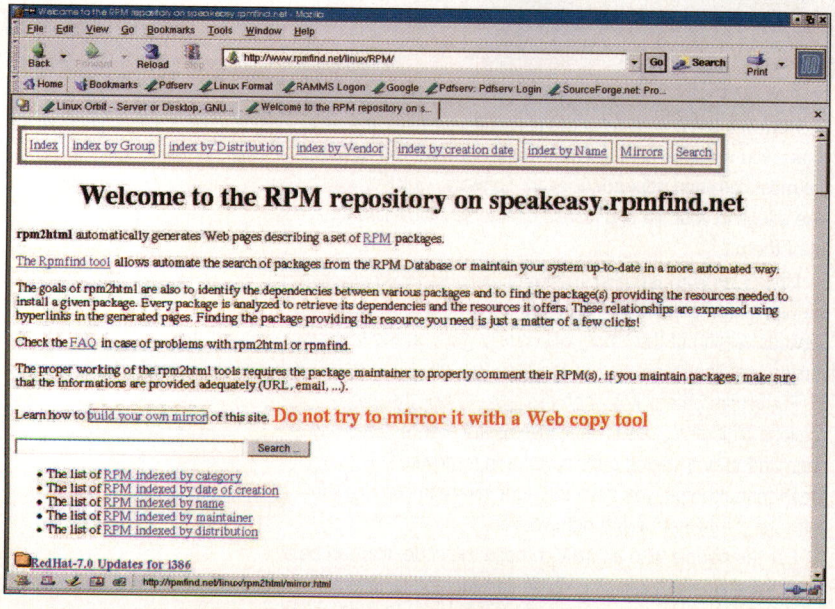
and then install it:

```
make install
```

This last step copies the executable to a general directory, in this case /usr/local/bin. If you want to install the software somewhere else, you can usually pass such information to the configure script.

When things go wrong

Well, that's how the compilation process is supposed to work, but what can you do when it doesn't? The first thing to do, if you


Welcome to the RPM repository on speakeasy.rpmfind.net - Mozilla
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://www.rpmfind.net/linux/RPM/
Go Search Print
Home Bookmarks Pdfserv Linux Format RAMMS Logon Google Pdfserv: Pdfserv Login SourceForge.net: Pro...
Linux Orbit - Server or Desktop, GNU... Welcome to the RPM repository on s...
Index index by Group index by Distribution index by Vendor index by creation date index by Name Mirrors Search
Welcome to the RPM repository on speakeasy.rpmfind.net
rpm2html automatically generates Web pages describing a set of RPM packages.
The Rpmfind tool allows automate the search of packages from the RPM Database or maintain your system up-to-date in a more automated way.
The goals of rpm2html are also to identify the dependencies between various packages and to find the package(s) providing the resources needed to install a given package. Every package is analyzed to retrieve its dependencies and the resources it offers. These relationships are expressed using hyperlinks in the generated pages. Finding the package providing the resource you need is just a matter of a few clicks!
Check the FAQ in case of problems with rpm2html or rpmfind.
The proper working of the rpm2html tools requires the package maintainer to properly comment their RPM(s), if you maintain packages, make sure that the informations are provided adequately (URL, email, ...).
Learn how to build your own mirror of this site. Do not try to mirror it with a Web copy tool
Search ...
The list of RPM indexed by category
The list of RPM indexed by date of creation
The list of RPM indexed by name
The list of RPM indexed by maintainer
The list of RPM indexed by distribution
RedHat-7.0 Updates for i386
http://rpmfind.net/linux/rpm2html/mirror.html

If you can't find particular libraries on your system or your magazine coverdiscs, a search engine like www.rpmfind.net can be useful.

Loki's *Setup* installer worked very well, though it would be too much to ask for every release of free software to go to the lengths of building a dedicated installer, which is why learning to compile software is so useful.

haven't already done so, is read the documentation. Most source distributions will come with a README and an INSTALL file. These often contain crucial information and useful options. There may be special configure options you need to use, or different things to do depending on your system. You should certainly read these files first if something does go wrong.

The most common failure is actually at the configure stage, when the configure script detects an out of date or missing library. Most common libraries will have been included with your distribution. If the library is simply out of date, you can usually update it from the update source for your distro.

For example, Red Hat maintains a page of updated RPM packages for most of their distros. Mandrake and SuSE both have online update tools too. You will also usually find source and binary packages for common libraries each month on *Linux Format* magazine CDs and the DVD in the Essentials folder.

Sometimes the configure script can be fooled if the library is present, but installed in an odd location, and therefore not in the list of paths that the library loading tool checks. Most distributions include the *locate* command or other tools which can be used to find files. If you are certain that you do have the correct version of the library, check with (for example):

```
locate libqt.so
```

which will return a path or list of paths where matching files can be found. Then you can take a look through the configure.log to find out how the configure script failed and if it was looking in the right place.

The configure script may well give you the option to specify the exact paths for particular libraries, and various other options. For example.

```
./configure -qt-libraries=/usr/lib/qt3/lib
```

Note that although sometimes you may have the libraries themselves, to compile software which makes use of them, you also usually need the development version, which will include header files the library was built with.

Summing up

Hopefully you will have found this short tutorial useful. It has given you some background on how to compile software and how the actual system works. By giving you some background information on this, you should be more able to solve any problems that you come across.

Trying to get some software to compile can be a rather frustrating process, but the vast majority of the source code you encounter will work fine provided you have the necessary libraries and the patience to read the instructions. ■

WebFusion®

seriously easy web hosting®

SALES HOTLINE **0870 333 9713**

for resellers

£50 PER MONTH OR £500 PER YEAR WebFusion Reseller

Reseller Includes:

- Unlimited Lightning-Fast UK Web Space
- **FREE and UNLIMITED** 24/7/365 phone and E-mail support
- 2 Virtual Servers included • Additional VS £25 per Year
- State-of-the-Art Data Centre • High Speed Connections
- Unlimited POP3 E-mail Boxes • Detailed On-line Help
- Regular Tape Backups • Additional Options Available
- **SAME HOUR SETUP**

All Your Accounts Have:

- NEW **10GB** Data Transfer (Bandwidth) per Month
- NEW MyServerWorld 10.0
- Unlimited E-mail Forwarding • Unlimited Mailing Lists
- Unlimited Autoresponders • Full CGI-BIN • Real Audio & MP3
- MIVA Scripting Language • MySQL Database
- Active Server Pages • PHP4 Scripting Language
- Graphical Access Statistics • SSH Telnet Access

£250 WebFusion E-Commerce EXTRA PER YEAR

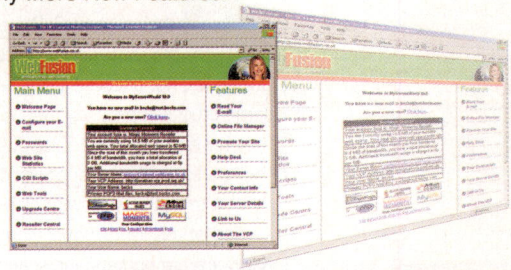
E-Commerce Only:

- All of the above plus
- Shopping Cart Solution, pre-installed, fully integrated with secure server and ready to run

"The new MyServerWorld 10.0 represents a quantum leap in control panel technology - nobody else can offer these features or functionality"

MyServerWorld 10.0

- All Our Usual Features Improved!
- Design Your Own Control Panel Skins 100% Branded to Your Company!
- Design Your Own Web Hosting Packages and Change Them in Real Time
- Full Customer Management System - Free of Charge!
- New Web Site Statistics With Graphical Reports
- Powerful On-line Mail Client
- Back Up Your Web Site & Manage MySQL Features On-line
- New Data Transfer Manager
- Web Site Link Checker & HTML Validator
- **500** Free Subdomains and **25** Free Accounts
- Many More New Features!



State-of-the-Art Data Centre

Our state-of-the-art data centre is located within our facility in Nottingham. It houses our network of over 4000 servers. With multiple connections to the net, fire protection and backup generators, we are in a position to offer you the reliability and stability you have been dreaming of!

24 / 7 / 365 **FREE** Technical Support

All WebFusion hosting packages come complete with 24/7/365 free telephone and e-mail technical support, plus an extensive online help and information system. Unlike our competitors, we believe in providing the very best after sales care and technical support.

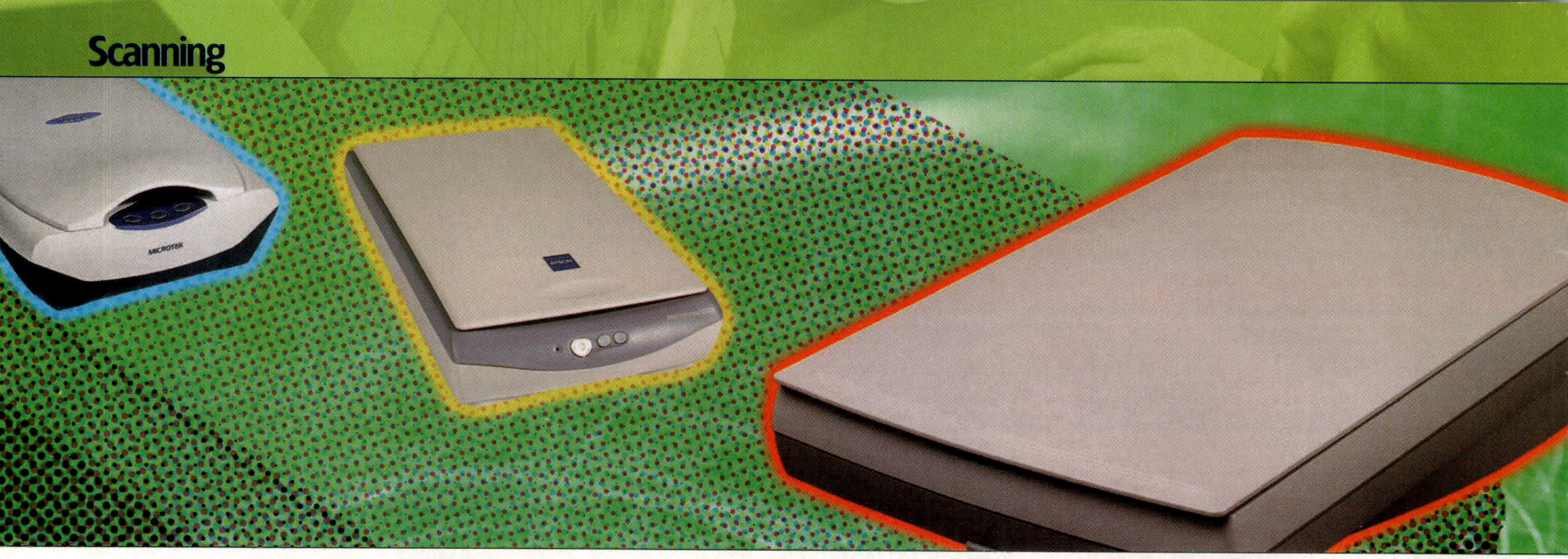
We welcome all major debit and credit cards. Prices exclude VAT @ 17.5%

e-mail us: sales@webfusion.co.uk

order on-line: www.webfusion.co.uk

supporting UK online for business

PART OF THE Host Europe GROUP



SCANNING with Linux

Richard Drummond unravels the mysteries of SANE to help you get your scanner working in Linux, and offers some tips on scanning quality.

What is it that's stopping most users adopting Linux wholeheartedly and removing Windows from their systems? The chances are most desktop users dual-boot, not because Linux lacks some crucial application, but because they have some piece of hardware that they think is either unsupported or too difficult to get to work with Linux.

Top of this list of troublesome hardware are printers and scanners which, considering how vital they are within today's modern digital office, can prove to be a considerable set-back on the desktop.

"Top of the list of troublesome hardware are printers and scanners – a set-back on the desktop for today's digital office."

Scanner support in Linux is provided by a piece of software called *SANE* which stands – optimistically, some might say – for *Scanner Access Now Easy*. *SANE* is a framework, an API for allowing applications to talk to image acquisition devices such as scanners. The job of controlling a particular device or class of device is done by an appropriate driver, known as a *SANE* back-end. The beauty of *SANE* is that it lets any application – or *SANE* front-end – talk to any supported device without worrying about the specifics of any particular device. In this respect, it's a lot like *TWAIN*, the system found in the Windows and Mac worlds, although *SANE* is much more flexible.

The ultimate question for any user is, "Is my scanner supported by *SANE*?" Alas, there's no quick answer here. *SANE* does provide back-ends for an impressive array of devices, including popular models from Agfa, Epson, HP and Mustek, but you'll have to visit the *SANE* website and check whether your device is supported in the current release.

If your scanner isn't listed and you're feeling adventurous, also look at the page listing third-party drivers which haven't yet been included in the official version of *SANE*. You should read any info regarding your scanner's *SANE* driver thoroughly before proceeding. This should tell you any kernel or other requirements it has, and what features are supported.

SANE is supplied with all the major distros these days, and if you're lucky, the kernel and the version of *SANE* installed on your system will already support your scanner. Load up your favourite *SANE* front-end and start scanning. If you're not so lucky, you may need to install a newer or patched version of *SANE*, or rebuild your kernel.

Driving in SANE

SANE back-ends are user-space drivers: they talk to your scanner at the application level. You still need a kernel driver to handle the hardware-

level communication between your system and the scanner. For SCSI scanners, this begins with a driver for your SCSI adaptor; for USB scanners, you need a working USB subsystem in your kernel. Fortunately, most distros will ship with a kernel which has all the necessary drivers supplied as modules. Your distro's hardware detection system may even load them automatically.

A quick way to check whether you have the appropriate kernel support for your USB or SCSI scanner is to use the *SANE* command **sane-find-scanner** (assuming you have *SANE* installed). This probes the USB and SCSI buses and reports any scanners that are found. Note, however, that just because a scanner is listed here, it doesn't necessarily mean it is supported by *SANE*; it just means that the kernel can access it. If your scanner is not reported then you may need to rebuild your kernel with the necessary drivers compiled in or built as modules. We haven't got the space to go into much detail here, but we can give you some pointers as to what kernel modules you may need to get your scanner working. Look at the table 'Kernel Drivers' for a summary. For more general help with building kernels see the Kernel HOWTO at **www.tldp.org**.

SCSI scanners should present no problems at the kernel level as they don't need any special kernel drivers, just a supported adaptor and the SCSI generic driver. Simply rebuild your kernel with the appropriate modules and ensure they are loaded at run-time.

USB scanners are a different story, because a USB scanner could actually be a SCSI scanner that has been given a USB interface. This is the case with the HP 5300C (a re-badged Avision scanner) and similar devices, which are supported by the *hpusbscsi* driver, and the Microtek X6USB, which is supported by the *microtek* driver. In both cases, theses devices will appear to the system as SCSI scanners, hence you'll also need the SCSI generic driver. Other USB scanners require the generic USB *scanner* driver.

To support any USB device, you'll also need appropriate host-controller support for your hardware, namely the *OHCI* or *UHCI* drivers, the *USB core* driver and so on. If you have a *USB hotplug* daemon running, this should take care of loading the appropriate drivers at run-time; if not, add the modules to your /etc/modules file.

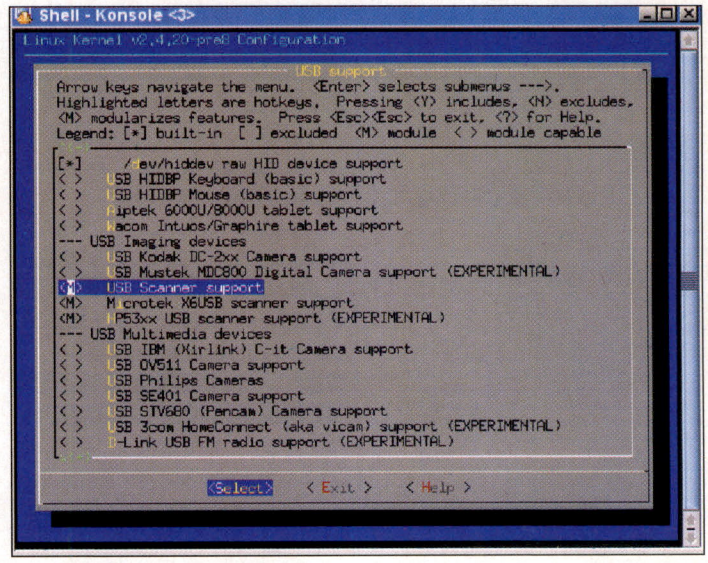
Parallel port scanners are less well supported by *SANE* and by the kernel. Like USB scanners, some parallel scanners may be SCSI devices in disguise. However, the official kernel doesn't include SCSI-over-parallel drivers, so you'll need to patch your kernel source to be able to build these (see **www.torque.net/parport/ppscsi.html**). Other devices may require support for user-space parallel drivers (this option can be found in the character devices page of xconfig), essential if you want to give non-root user access to the scanner. Whichever type of device you have, you'll obviously need a parallel port driver in your kernel.

Finally, IEEE1394 (Firewire) devices are now beginning to be supported by *SANE*. These require a kernel with IEEE1394 support and the *SBP2* (Serial Bus Protocol) driver. This provides a SCSI interface, so again you'll need the SCSI generic driver.

Scanner access

The final piece of the puzzle as far as kernel issues go is to ensure that your system has the correct device files installed, so that *SANE* can make use of the kernel drivers you have just built. If your system uses DevFS, then the drivers will automatically create their own device nodes as required. Otherwise, you may need to install them manually.

Devices which use the SCSI generic interface, whether they are real or emulated SCSI devices, require a device file of the form /dev/sgX, where X depends on the ordering of SCSI devices attached to your system. (Have a look in /proc/scsi/scsi. If your scanner is the fourth device, it will be accessed via /dev/sg4.) An appropriate range of /dev/sgX files should already be present on your system, although on some systems these might be ordered by letter rather than number (in other words, you'll have /dev/sgd rather than /dev/sg4).


Shell - Konsole <3>
Linux Kernel v2.4.20-pre8 Configuration
USB support
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help.
Legend: [*] built-in [ ] excluded <M> module < > module capable
[*] /dev/hiddev raw HID device support
< > USB HIDBP Keyboard (basic) support
< > USB HIDBP Mouse (basic) support
< > Aiptek 6000U/8000U tablet support
< > Wacom Intuos/Graphire tablet support
--- USB Imaging devices
< > USB Kodak DC-2xx Camera support
< > USB Mustek MDC800 Digital Camera support (EXPERIMENTAL)
<M> USB Scanner support
<M> Microtek X6USB scanner support
<M> HP53xx USB scanner support (EXPERIMENTAL)
--- USB Multimedia devices
< > USB IBM (Xirlink) C-it Camera support
< > USB OV511 Camera support
< > USB Philips Cameras
< > USB SE401 Camera support
< > USB STV680 (Pencam) Camera support
< > USB 3com HomeConnect (aka vicam) support (EXPERIMENTAL)
< > D-Link USB FM radio support (EXPERIMENTAL)
<Select> < Exit > < Help >

Depending on your scanner model and your distro, you may need to rebuild your kernel in order to get scanner support.

To create them yourself, use a command of the following form:

```
mknod /dev/sgX c 21 X
```

It is common practice to create a link from the device /dev/scanner to the actual SCSI generic device, making the scanner quicker to find.

```
ln -s /dev/sg4 /dev/scanner
```

If, on the other hand, your scanner uses the generic USB scanner module, you need to create a /dev/usbscanner file. Do this with

```
mknod /dev/usbscanner c 180 48
```

Don't create a link from /dev/scanner to /dev/usbscanner, though, because this will just confuse *SANE*. If you have more than one USB scanner then you can make nodes of the form /dev/usbscannerX, with X replaced with the number of the scanner. Minor

Kernel Drivers

Which driver do I need for my scanner?

DRIVER	CONFIG OPTION	DEVICE NODE	NOTES
sg	CONFIG_CHR_DEV_SG	/dev/sgX	SCSI generic driver. Required for all SCSI and emulated SCSI scanners.
scanner	CONFIG_USB_SCANNER	/dev/usbscanner	Generic USB scanner.
hpusbscsi	CONFIG_USB_HPUSBSCSI	-	Supports USB Avision scanners such as the HP 5300C. Requires sg.
microtek	CONFIG_USB_MICROTEK	-	Supports Microtek X6USB and similar. Requires sg.
ppdev	CONFIG_PPDEV	/dev/parport	Support for user-space parallel drivers. May be required for some parallel-port scanners.
sbp2	CONFIG_IEEE1394_SBP2	-	IEEE1394 serial bus protocol. Supports Firewire scanners. Requires sg.

numbers 48 to 63 are reserved for USB scanners (you can have up to 16 if you want), so 48 is the first scanner, 49 the second and so on.

Last but not least, you need to ensure that your scanner's device file has permissions set so that it is accessible by non-root users. The typical way to do this would be to set its ownership to the group 'scanner' and add any user who is permitted to use the scanner to this group. *E.g.*

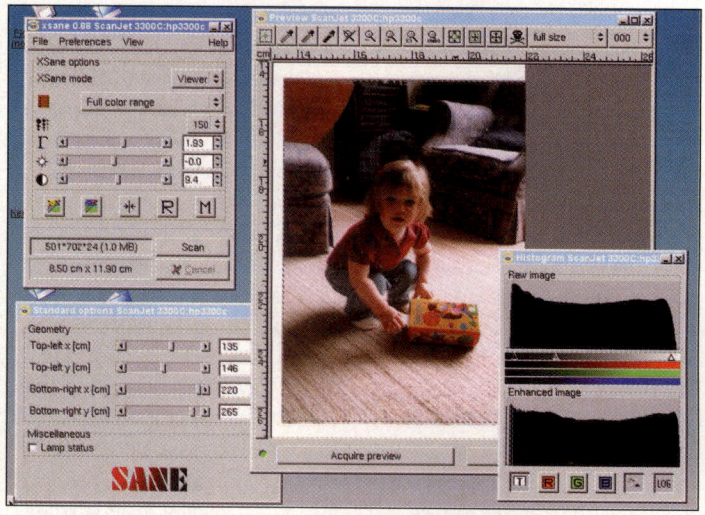
```
chgrp scanner /dev/sg4
chmod g+rw /dev/sg4
```

Note that you need both read and write access for proper scanner operation.

Using SANE

If you don't already have *SANE* installed, if you need a newer version to support your scanner or if you want to add a third-party driver, then you'll probably want to build *SANE* yourself. Fortunately, this is straightforward. The *SANE* package is supplied as two tarballs, called *sane-backends* and *sane-frontends*. The former contains the *SANE* drivers, library and tools and is all you really need, unless you really want the *xscanimage* and *xcam* applications supplied in the frontends package (there are better *SANE* front-ends available).

To build the backends package, it's a case of unpacking the tarball and performing the usual **configure/make** incantation. If you are adding a third-party driver, you should do that first, and most add-on drivers will supply instructions on how to do this or even supply a script to do the job for you. *SANE* doesn't have any peculiar build requirements, although some parallel drivers may need the *libieee1284* package (see **http://cyberelk.net/tim/libieee1284/**) and USB drivers may



***XSane* is currently the most powerful *SANE* front-end, but the competition are catching up fast.**

SCANNING TIPS!

Find your scanner sweet spots!

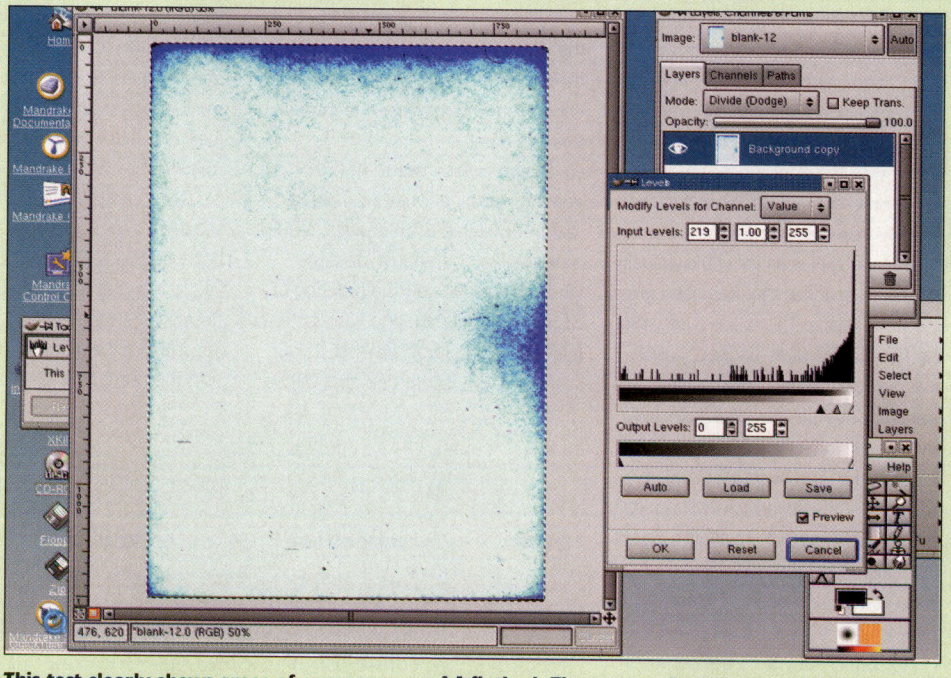
Many people don't realise that scanners, like many other devices that interface to an analogue world, have a sweet spot. Due to the way scanners, particularly colour ones, work, there are always some areas of the scanning bed that give better reproduction than others, and some where system noise, for want of a better term, interferes with the results.

This can make a profound difference, especially if you realise that the 'noisy' areas are most often around the edges of the scanning area – where most people like to line up their photos or whatever against a clean edge.

Sweet spots can vary depending on the resolution you are scanning at, and whether you are using greyscale or colour mode. If you want to be really picky, you could prepare a sweet-spot 'map' for different resolutions and colour depths.

So, how can we easily make a sweet spot 'map' in the first place? What you need to do is scan a pure surface (and it helps to clean the glass first). White is usually best. It is important that you scan a clean, white surface. A piece of paper gives a rough idea, but good quality, thick uncoated card is better. To eliminate the possibility of imperfections in the card giving false readings, you can turn it round and do another. You can then manipulate the image to include only the common areas of interference.

When you have the scan, you may already be able to see some coloured areas. Load the image into *GIMP* and use the Levels or Curves tool to maximise the contrast, which will reveal the effect more clearly. When scanning large objects which have to go over noisy areas of the scanning bed, it's possible to use the map you created to correct the image (say, as a 'Divide' layer in *GIMP*), but it can take a lot of work to actually improve an image.



This test clearly shows areas of concern on an A4 flatbed. The upper edge corresponds to the open end of the lid, so a possible cause here is light leaking in. The dead zone on the side could be due to a number of factors. Note that the blue cast to most of these areas indicate that the blue head is less susceptible to the problem.

need *libusb* (see **http://libusb.sf.net/**). You should perform the **./configure --help** command to see what options apply to you before building. You should at least supply the **--prefix** option to specify the install location, which will typically be /usr or /usr/local. If you have an existing *SANE* install, it's safest to remove that before installing a new version.

To check that your scanner's driver has been successfully built and installed, execute the command

```
scanimage -L
```

This will try all the installed back-ends and report any devices found. In my case, this finds my USB ScanJet 3300C and returns with the line

```
device `niash:hp3300c' is a
Hewlett-Packard ScanJet 3300C
flatbed scanner
```

Here **niash:hp3300c** is *SANE's* device name for my scanner. *SANE* uses a simple naming scheme for devices. The first part, in this case **niash:** is the name of a *SANE* back-end. The second part is driver-specific and will usually be the path to the device file via which the scanner can be accessed, like /dev/usbscanner0 or /dev/sg2. When using *SANE* applications, if you don't specify a default device to use, *SANE* will search for one, which significantly slows down application start up.

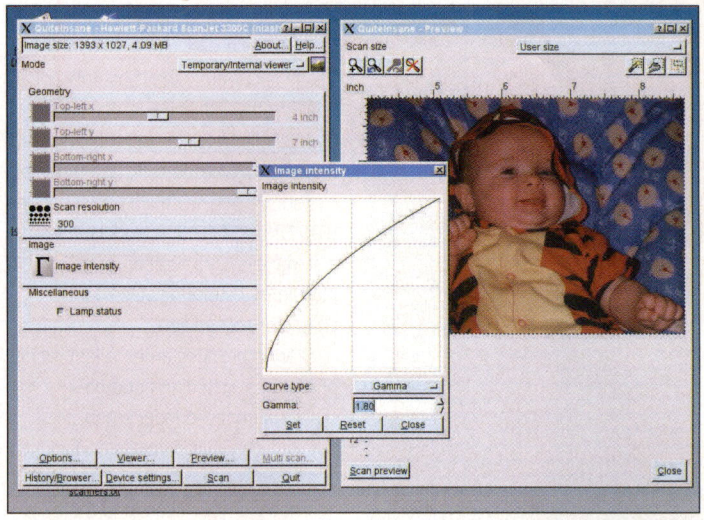
Now you know how to access your device, you can test out the driver by supplying the **-T** switch to scanimage. This will perform a number of tests and report whether the driver passes or fails. For example:

```
scanimage -T -d niash:hp3300c
```

If this fails, consult the *SANE* FAQ and your *SANE* driver's manual pages. The latter should contain additional information about the driver and what options its supports. (Most *SANE* back-ends can use a file in /etc/sane.d/ called <back-end>.conf for storing default settings.) If all goes well, you're ready to use a full-blown *SANE* front-end and do some proper scanning.

SANE front-ends

The *SANE* package includes several *SANE* front-ends, applications which make use of *SANE* drivers for image acquisition. These include the command line *scanimage*, which we have been using, and the GUI-based *xscanimage* and *xcam*. The first of these is useful for testing and batch purposes, but not up to everyday scanning, while *xscanimage* and *xscan* are too basic to be really useful. Thankfully, there are several good *SANE* applications available.

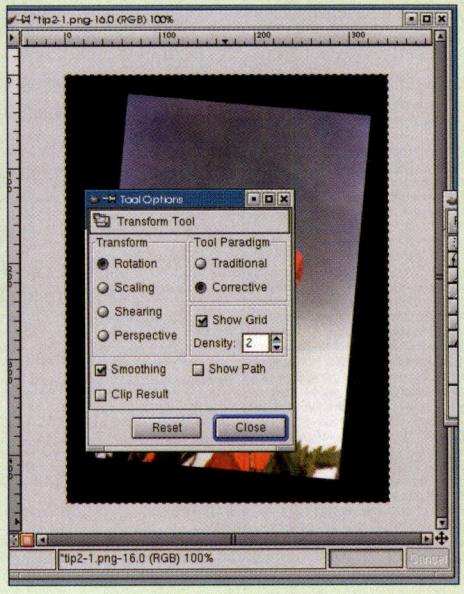


***QuiteInsane* is a new *SANE* front-end that looks promising, although much work needs to be done yet.**

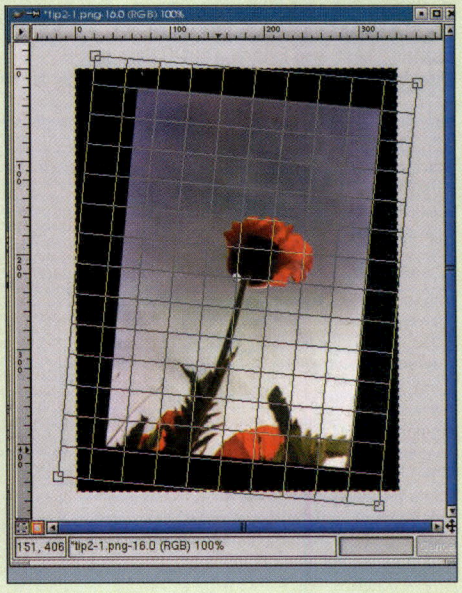
Correcting wonky scans

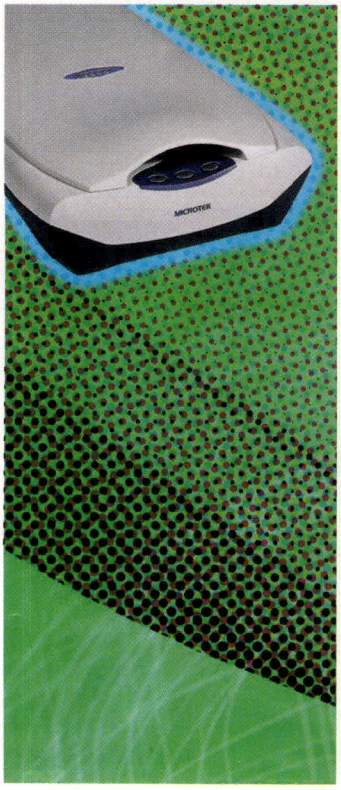
Sometimes when scanning pictures, almost inevitably if you are keeping away from the edges, you end up with a crooked scan. *GIMP* is here to help you though! We have exaggerated a wonky scan here in our example.

First of all, double click on the Transform tool in the *GIMP* toolbox to get up the tool setting dialog box. Now change the "Tool Paradigm" setting to Corrective. This means that the tool is set up to correct the transformation effect which you set.



Now use the Transform tool and rotate the grid to square up with your scanned image. The Transform function will correct this angle, making your scan straight again. The same technique can be used to straighten parts of a photograph that were wonky to begin with.





XSane

www.xsane.org

Xsane is currently the most functional *SANE* front-end, and the one most people will prefer to use. It provides a friendly, *GTK*-based GUI, and the current version will build against either *GTK-1.2* or *GTK-2.0*. It works either as a stand-alone application or as a *GIMP* plug-in and supports *JOCR* for character recognition.

XSane provides an excellent range of features, including good pre-scanning image-enhancement tools. You can easily crop, rotate and zoom in the preview window to control the geometry of your scan, and there's a histogram function for the quick and accurate setting of image intensity.

XSane saves images in a variety of formats, prints images and sends them via fax or email. Its built-in image viewer is rather primitive, however, and has only basic post-scan image processing tools, only two filters (blur and de-speckle) and no undo function.

QuiteInsane

http://quiteinsane.sourceforge.net/

QuiteInsane is a fairly new *SANE* front-end and still in an alpha state. It has a pleasant *Qt*-based GUI which will build against either *Qt2.x* or *Qt3.x*. Its preview functions are less easy to use than *Xsane*'s, and it's noticeably lacking a histogram tool. Instead, there are various functions for setting the intensity/value curve, including a classic gamma function. It doesn't update the preview on the fly, so it's difficult to set this appropriately, however.

QuiteInsane makes up for these shortcomings by supplying a range of tools and filters to enhance your image post scanning. This includes colour and intensity controls, sharpen and blur filters and effects such as posterize. Stability can still be a problem, though.

Kooka

www.kde.org/kooka

Kooka is the *SANE* front-end featured in KDE 3.x, although versions for KDE2.x do exist. It is built on the KDE *libkscan* library, which can provide scanner services to all KDE applications, and is used in the latest version of *KOffice*. *Kooka* has a very configurable user interface and great integration with other KDE apps, but I've never had much success with it.

In terms of scan control features it looks about on a par with *QuiteInsane*, but as I've never been able to get it to do a preview scan, I can't really comment. Its a promising-looking project, though, so stay tuned.

Automatic reading?

OCR or Optical Character Recognition is any process for converting printed pages into text in the computer's memory. Not surprisingly this is difficult to do, involving complex image-processing techniques, and early OCR packages gained a poor reputation.

In recent years, OCR software has improved immeasurably, and processional OCR software is now commonly used commercially for archiving documents.

In the Free Software world, there's at least one useful OCR program, and that's *JOCR*. (Confusingly, this application used to be known as *GOCR* – and the program executable is still called **gocr** – but there's another, separate project called *GOCR*.)

JOCR is a command-line tool, although a simple *GTK* front-end is also included in the package and all the popular *SANE* front-ends can integrate with *JOCR*. To use *JOCR* from the command line you simply need to supply an image file for it to decode.

```
Gocr scan.pnm -o scan.txt
```

It only works with PNM, PGM or PPM images, but with other image formats you can use tools in the NetPBM package to convert them first. For JPEG images, the *djpeg* tool from the *libjpeg* package is also useful. For example:

```
djpeg -gray myscan.jpg | gocr -o out.txt
```

A useful feature in recent versions of *JOCR*, if you need to scan a lot of pages which use the same font, is the ability to create a dictionary mapping glyphs it cannot recognise to the appropriate characters or strings.

You can run **gocr** in an interactive mode – with the switch **-m 130** – and it will prompt you to enter values for any characters it fails to recognise, with the option of storing those values in the dictionary.

JOCR is a work-in-progress and so the results can be variable. It is rather slow, uses a lot of memory and has problems with complex page layouts, serif fonts, italics and ligatures. It's definitely a project that's worth supporting, though.

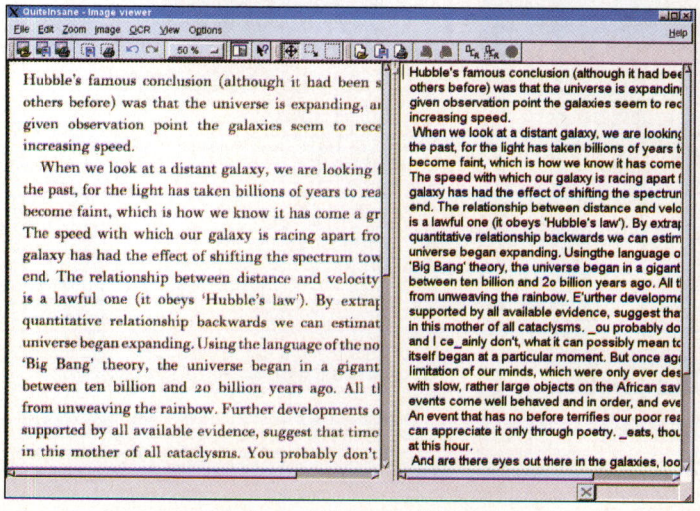
Not just for scanners

SANE provides back-ends for a wide range of image acquisition devices – it's not all just about flatbed scanners, you know.

The *v4l SANE* back-end supports any video capture device with a kernel *Video4Linux* driver, including TV cards and web-cams. Here, the *SANE* device name specifies the device file to use – for example, v4l:/dev/video0.

If that's not enough, the *gphoto2* driver lets you access any digital camera supported by the *Gphoto2* library (see **www.gphoto.org** for more information). With this driver, you must specify the parameters of the device to be used in the /etc/sane/gphoto2.conf file. ■

QuiteInsane integrates well with _JOCR_ for OCR, and lets you view scanned text and interpreted text side-by-side.

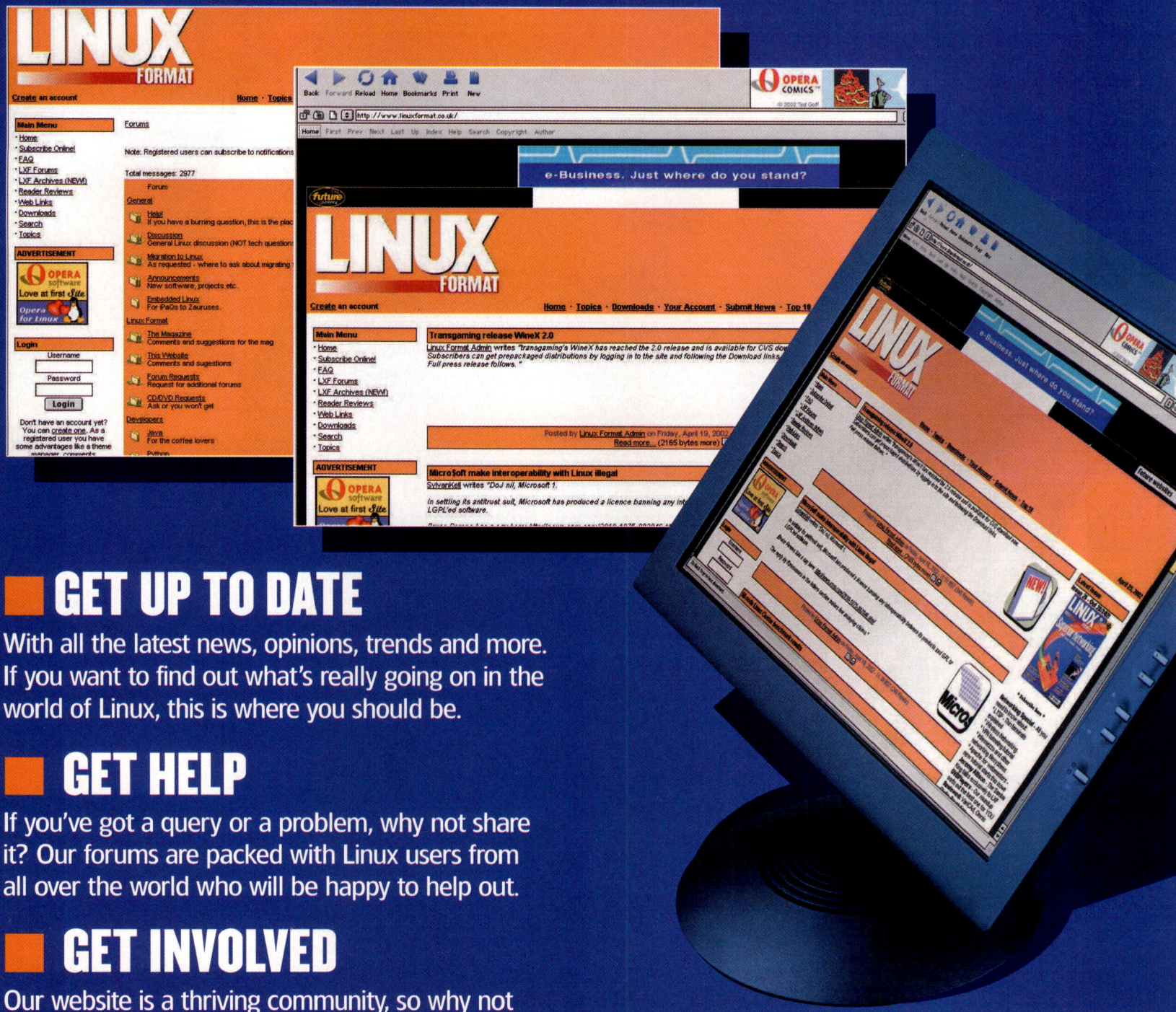

QuiteInsane - Image viewer
File Edit Zoom Image OCR View Options Help
Hubble's famous conclusion (although it had been s
others before) was that the universe is expanding, a
given observation point the galaxies seem to rece
increasing speed.
When we look at a distant galaxy, we are looking
the past, for the light has taken billions of years to rea
become faint, which is how we know it has come a gr
The speed with which our galaxy is racing apart fro
galaxy has had the effect of shifting the spectrum tow
end. The relationship between distance and velocity
is a lawful one (it obeys 'Hubble's law'). By extrap
quantitative relationship backwards we can estimat
universe began expanding. Using the language of the no
'Big Bang' theory, the universe began in a gigant
between ten billion and 20 billion years ago. All th
from unweaving the rainbow. Further developments o
supported by all available evidence, suggest that time
in this mother of all cataclysms. You probably don't
Hubble's famous conclusion (although it had bee
others before) was that the universe is expandin
given observation point the galaxies seem to rec
increasing speed.
When we look at a distant galaxy, we are lookin
the past, for the light has taken billions of years t
become faint, which is how we know it has come
The speed with which our galaxy is racing apart f
galaxy has had the effect of shifting the spectrun
end. The relationship between distance and velo
is a lawful one (it obeys 'Hubble's law'). By extra
quantitative relationship backwards we can estim
universe began expanding. Usingthe language o
'Big Bang' theory, the universe began in a gigant
between ten billion and 20 billion years ago. All t
from unweaving the rainbow. E'urther developme
supported by all available evidence, suggest tha
in this mother of all cataclysms. _ou probably do
and I ce_ainly don't, what it can possibly mean to
itself began at a particular moment. But once ag
limitation of our minds, which were only ever des
with slow, rather large objects on the African sav
events come well behaved and in order, and eve
An event that has no before terrifies our poor rea
can appreciate it only through poetry. _eats, thou
at this hour.
And are there eyes out there in the galaxies, loo

SANE drivers

Where to find information

SANE home page
www.mostang.com
SANE back-ends
www.mostang.com/sane/sane-backends.html
Third-party back-ends
www.mostang.com/sane/sane-backends-external.html
Official _SANE_ FAQ
www.xs4all.nl/~ljm/SANE-faq.html
USB scanner resources
www.buzzard.org.uk/jonathan/scanners-usb.html
Parallel scanner resources
www.buzzard.org.uk/jonathan/scanners.html
HP OfficeJet project
http://hpoj.sourceforge.net/
Epson scanner drivers
http://khk.net/sane/

www.linuxformat.co.uk



GET UP TO DATE

With all the latest news, opinions, trends and more. If you want to find out what's really going on in the world of Linux, this is where you should be.

GET HELP

If you've got a query or a problem, why not share it? Our forums are packed with Linux users from all over the world who will be happy to help out.

GET INVOLVED

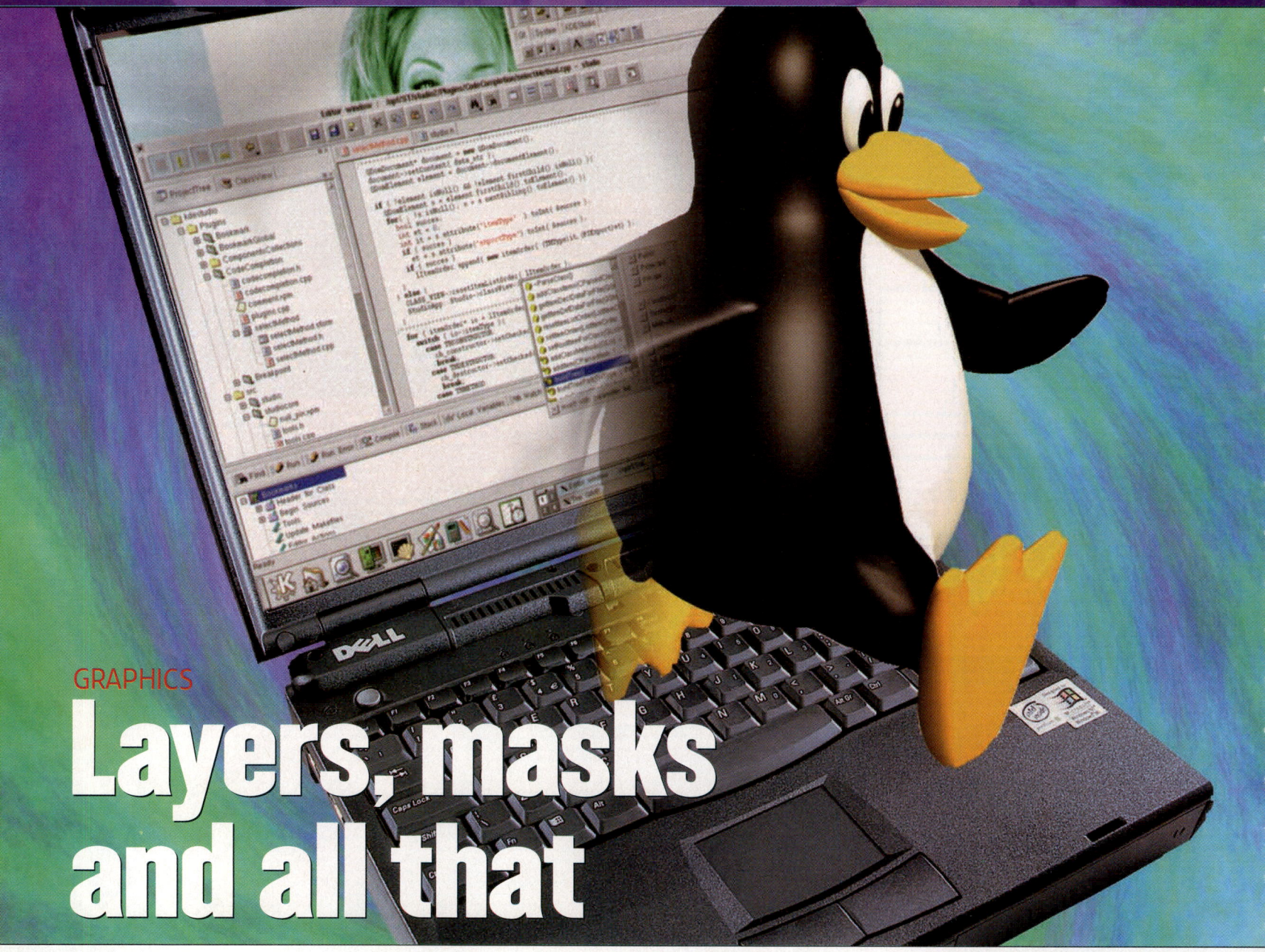
Our website is a thriving community, so why not contribute your own articles and ideas and give something back to the Linux world?

GET ONLINE!

Fire up your modem and surf on into the Linux-friendly website of the UK's best-selling Linux magazine.

LINUX FORMAT

The UK's best-selling Linux magazine



GRAPHICS

Layers, masks and all that

Nick Veitch introduces you to a small slice of the power of The GIMP when it comes to creating composite images.

With this guide to getting the most out of *The GIMP*, we're going to take a look at creating a composite image from several different elements. The reasons for doing this are simple: since the advent of layers, creating composite images has been made a lot easier, but also allows for a great deal of creative flexibility. You can do things with layers, adjust effects and alter sequences, which enables you to continue tweaking and changing the image until you're happy with it. Once you understand how to create composite images quickly and easily, you'll be able to be a lot more creative with your artwork. The procedure we'll follow here isn't the only one, but it does also take in the key skills of making selections and using alpha-channels and masks.

STAGE 1: Creating a cutout

To create a composite image, you obviously have to start out with some basic elements. Whatever source these images are, you're more than likely to need to separate them from their backgrounds to create an often irregular-shaped element, which can then be used in the final image. In the graphics industry, this is known as a cutout.

The first step on the way to a cutout is to select the area you wish to cut out. There are a number of ways to do this with *GIMP*, depending on the shape of the image you want to cut out and the background it's on. The simplest object is a square box – you can easily make the selection with the rectangular selection tool (the first tool on the *GIMP* toolbox).

For slightly more complicated objects, you might be able to get away with using the rectangle and ellipse selections. You can combine selections by holding down the **Shift** key while dragging out the new area, and subtract them by holding down the Control key and dragging out the area to be removed.

If the object is on a perfect or near perfect background, you can use the Fuzzy Select tool. It resembles the magic wand tool from PhotoShop and has a similar effect, selecting contiguous areas of the same or similar colour. If you had a red splat on a blue background, this would probably be the tool to use. The Lasso tool isn't really that useful for this sort of work, unless you

have an incredibly steady hand and can draw with a mouse. It can be useful for adding and removing small isolated areas of selections though, using the **Shift** (add) and **Control** (subtract) key modifiers.

The next tool worth considering is not a selection tool at all, but the path-drawing tool (it looks like an inkpen with a line and a dot on it). This is for drawing bezier-curve paths, but can be used to create cutouts, too. Use the bezier tool to create curved paths around an object: click for each point, holding down and dragging the mouse to control the curve, then close the path by making the end point on top of the start point and double clicking inside the enclosed area.

If you now click on the Paths tab in the Layers, Channels and Paths dialogue box, you'll see the path you've made and some extra options at the bottom, one of which is Convert Path to Selection. Paths might seem more complicated than some of the other tools, but they're the best way to cut out clean shapes from any background – especially if they're in any way complicated. For a quicker way, though, there's one last tool...

The Intelligent Scissors tool offers an easy and fast way to do quite good cutouts. You use it by clicking on points around the image you are cutting out, and the software will pick up the 'edge' itself. Obviously, this works best if the image has a readily visible edge! A word of warning though: this tool is still in development and can be erratic.

Let's take a look at some examples. Firstly, we have a simple box on a nice background. There are a few shadows on the background, but there are still some nice clean edges here. This could be a contender for any of the methods discussed, but probably the fastest tool for the job is the Path method. Why? Because the edges are pretty straight, so it's just a matter of putting down a few points and you're done. The Intelligent Scissors tool might be able to do the job, but it could also make the edges appear a bit raggedy.

STEP1: Select the Path tool (the inkpen with the line, remember?) from the toolbar. Now start at the top left and lay down six straight points in the corners of the image. Just click and release to define the points; don't drag out any curves or anything yet. Click once again on the initial point to close the path, and when you move the cursor over the box, it'll change to a square.

STEP 2: In the Path dialogue window, you'll see a little outline of the path you've created. Click on the button with a wiggly line on it, on the row at the top. This is used to edit your points – you can add and remove them with the other buttons. The only bit we need to worry about is the curved edge at the top right. Click on the front corner point and the curve handle will appear (you may have to drag them out from the point to see them clearly). Hold down the **Shift** key to adjust one handle at a time and fiddle with them until the curve fits. Depending on where your points are, you may want to adjust the handles on the back corner too.

STEP 3: Take a final look around to make sure none of the background is within the path – it doesn't matter so much if you cut off some of the box. Then you can convert the path to a selection by clicking the Path to Selection button (the grey circle along the bottom), and hey presto!

At this point it's usually a good idea to save the selection, because chances are you'll be wanting to use it. From the main menu, choose Selection>Save to channel. If you ever need to restore it, click the Channel tab on the Layers, Channels and Paths dialogue box, right-click on the channel and select Channel to Selection from the pop-up menu.

STAGE 2: Creating a Mask

Now you have the selection, all you need to do is turn it into a mask for the image. The mask is used to literally mask off the areas of the image that aren't part of the object, and makes those areas of your image transparent.

As you have the selection of the part of the image you want to keep, it's fairly simple to turn this into a mask. But why bother? After all, you could just use the selection to delete everything else in the image. That's true, and it's often a very useful first step because it causes less confusion that way. But even if you do delete the rest of the image, a mask layer is very useful when it comes to compositing the image.

STEP 1: In this example we've got a different image, which we used the Intelligent Scissors selection mode on. Although this one's on a white background, we still want to get rid of it, otherwise areas of white might show through whatever background we use. Invert the selection, select Feather and choose a feather radius of two pixels (depending on the image). Now choose Edit>Cut from the menu to cut that area from the image. The feathered background leaves our cute cutout with a friendly, soft edge.

STEP 2: Go to the channels menu and reload the selection by selecting the channel you created earlier and clicking on the Channel to Selection button. Now click on the Layer tab and right-click on the current layer until the menu pops up. Select

»

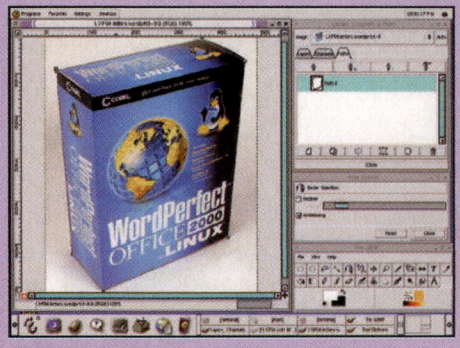
Versions

The basic techniques which are covered in this guide relate to all versions of the GIMP software older than version 1.0. You'll find the latest version of *GIMP* on the attached coverdisc.

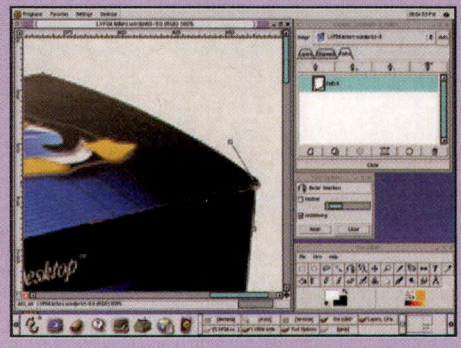
For more information, or to download it for yourself, check out the website at www.gimp.org

STAGE 1 – Creating a cutout

Cutting your image from the background



Create a rough path using the Bezier tool, making sure you've covered the main area.

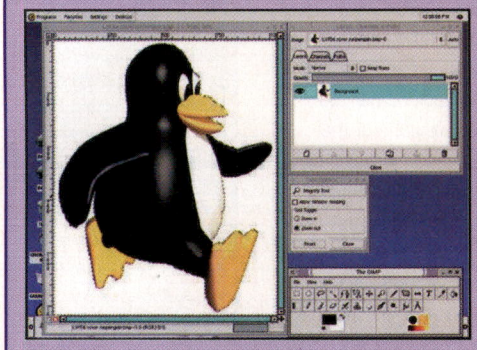


Tweak the curve to fit the box area precisely by editing the points you created.

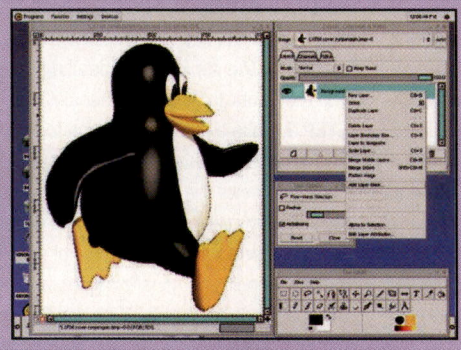
Finally, save the path and convert it into a selection from the path dialogue window.

STAGE 2 – Creating a mask

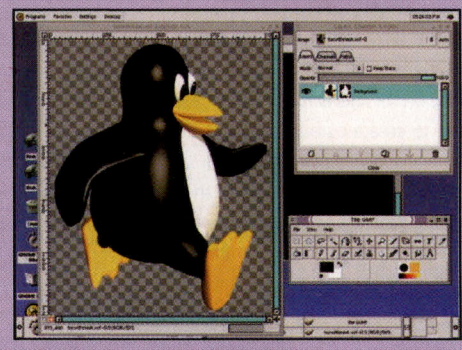
Using a mask to hide the background



With the selection on a soft object, you can feather it before killing the background.



With the selection inverted, add an alpha channel to the current layer, then add a mask.



Select the mask for the object layer and fill the selection with black to kill the background.

« Add Alpha Channel to create an alpha channel for this image. Right-click on the layer again and select Add Layer Mask from the menu. Select Fill with White from the requestor – it doesn't matter, we'll create our own mask from the selection.

STEP 3: Now the layer has a thumbnail of the image, plus a blank square next to it. This is the thumbnail of the layer mask. You can select between working on the mask and working on the image by left-clicking on the appropriate square. The currently selected item will be highlighted with a white surrounding box.

You should still have the area around the object selected, so right-click on the main image and choose Edit>Fill with FG (foreground) colour, which is black by default. The surrounding area will fill with black on the mask, making it transparent. The penguin will now have a chequerboard transparent background.

STAGE 3: Compositing

Now we've actually got some objects, we can start putting them together. Compositing using layers is quite easy, but also fairly memory and processor hungry so make sure your images aren't too huge or compositing could take you all day!

STEP 1: To start with, you'll need two open images. Sadly, *GIMP* doesn't have an effective way of merging layers from one image with layers of another, except by using the rather cunning drag-and-drop technique. With both images open, bring one to front so you can see it. On the Layers dialogue, use the image selector at the top to select the other image. Now click and drag that layer onto the open image window of the other image and a new layer will be created!

STEP 2: If you try the same trick with the mask, you'll find yourself with just another layer. To transfer the mask from the previous image, you'll need to use *GIMP*'s buffer system. Make sure you have the mask selected (click on it in the layer window) and select all of it (use the **Control-A** shortcut). Now, right-click on the image window, select Edit>Buffer>Copy Named and enter a name for the buffer (like 'tux mask').

STEP 3: Now go to the composite image. In the Layer menu, add a Mask layer and select it as before. Right click on the image and select Edit>Buffer>Paste Named. From the list that appears, select the tuxmask buffer that you created in the previous step. The mask will now be loaded as a floating selection. Just left-click on the image and the mask will automatically be applied to Tux.

STEP 4: Now you have two objects on two layers. You can move the top object around by selecting that layer and using the Move tool to reposition it wherever you want.

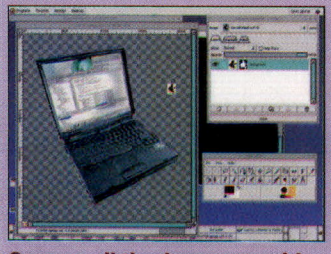
You can also use the image mask to create interesting effects. For starters, select the Tux layer and then right-click on the layer and select Mask To Selection. Now select the gradient tool and draw a line from left to right in the selection. You'll see Tux can be made to materialise!

Final word

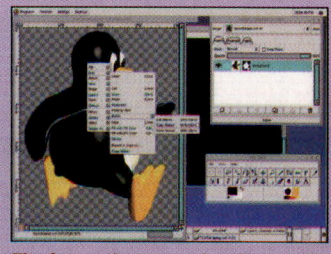
There's obviously a lot more that you can do with masks and layers, and probably several thousand ways for you to create interesting composite images. We hope this tutorial has shown you the basics, so you'll be able feel confident enough to go on and be even more experimental. ■

STAGE 3 – Compositing

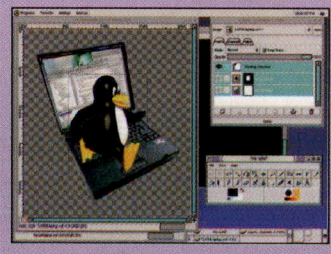
Bringing your images together



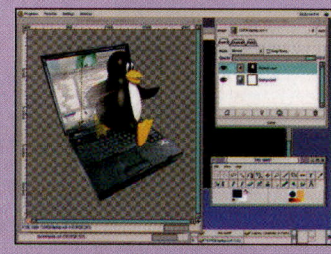
Open up all the documents with your object cutouts.



The image layer can be drag-and-dropped from the layer dialogue.



Copy the mask layer using *GIMP*'s named buffer system.



Apply a bit of imagination to the mask for a more creative image!

LINUX CONNECTIONS

Connecting to USB cameras

Say cheese! Nick Veitch coaxes images from the next generation of USB digital cameras.

Many of the new range of USB cameras on the market do not conform to the traditional model of serial camera communications. In the past these devices have used their own serial protocols and custom commands to communicate with a host over a serial link. Many of these cameras are already ably supported by *gphoto*.

However, a growing number of USB devices (but not all!) do not use traditional communications. Instead these devices take photographs and store them on a Flash card or Smartcard, which is treated solely as a storage device. This device is then simply represented on the USB bus as a mass storage device – the host computer communicates with the Flash card as a storage device.

This is a great boon to the ability of Linux to support these devices. Their appearance on the Universal Serial Bus means the usb-storage module, which also deals with USB CD/CDRW devices, hard drives and other storage devices, can simply address the cameras contents as though it were accessing a hard-drive. No complicated protocols or command sets; you just need to mount the device and you can copy the pictures off it.

USB support

The first thing to do is check that your kernel supports USB. Many older distributions used patched kernels to provide some USB support, but it is certainly much improved in later releases from 2.4 onwards.

Recent distros like SuSE 8, Mandrake 9.0 and later will come with a kernel supporting mass-storage devices. As this is a basic class-level support, rather than a particular device driver (like some of the Kodak DC series) it is most likely to be enabled and working in your kernel if you have a modern distribution.

It is also likely that you have the usbd service running. This monitors the USB ports for devices that are removed or become available. Obviously, this is handy for something like a camera. Check that the usbd daemon is running on your system (you'll need to be logged in as root to do this):

```
ps x | grep usb
```

The next thing you need is the usb-mass storage module. Any USB-enabled kernel will probably already have this compiled as a module. If your usbd service wasn't running you may have to install it yourself though.

Connect up the camera via a USB port and enter:

```
modprobe usb-storage
```

This will start the usb-storage module. To check that the device is connected and that the module has loaded properly, you can check out the /proc filesystem entries:

```
less /proc/bus/usb/devices
less /proc/modules
```

The usb-storage module will map the device using pseudo-SCSI support, so it will actually appear as a SCSI device on your Linux box. If you have no SCSI interface, the device is likely to be sda, otherwise it could be sdb or something else depending on which distro you use, and what SCSI devices you have.

Now you need to create a mountpoint for the camera-drive:

```
mkdir /mnt/camera
```

should be fine.

Now we can test that the mount will actually work, by using the mount command manually:

```
mount -t vfat  /dev/sda1/  /mnt/camera
```

The mount command here is using the vfat filesystem. This tends to be the filesystem universally used on USB cameras which are supported as drives (for some reason).

Once the device is mounted, you should be able to use it just like any other drive, although it will be a little bit slower. The camera may use a special directory for storing the picture files, which varies from camera to camera. Try **ls /mnt/camera** to see any directories on the device.

Mountlist entry

When you are sure that the device works, you'll probably want to set up a mountlist entry for it in your /etc/fstab file. If you have supermount or automount you'll probably want to set it up using that. Using your favourite text editor, open the /etc/fstab file and add a line:

```
/mnt/camera   /mnt/camera/  supermount
dev=/devsda1,fs=vfat,defaults,noauto 0 0
```

or you could manually mount and unmount the drive with:

```
/dev/sda1/ /mnt/camera  vfat  defaults 0 0
```

In use

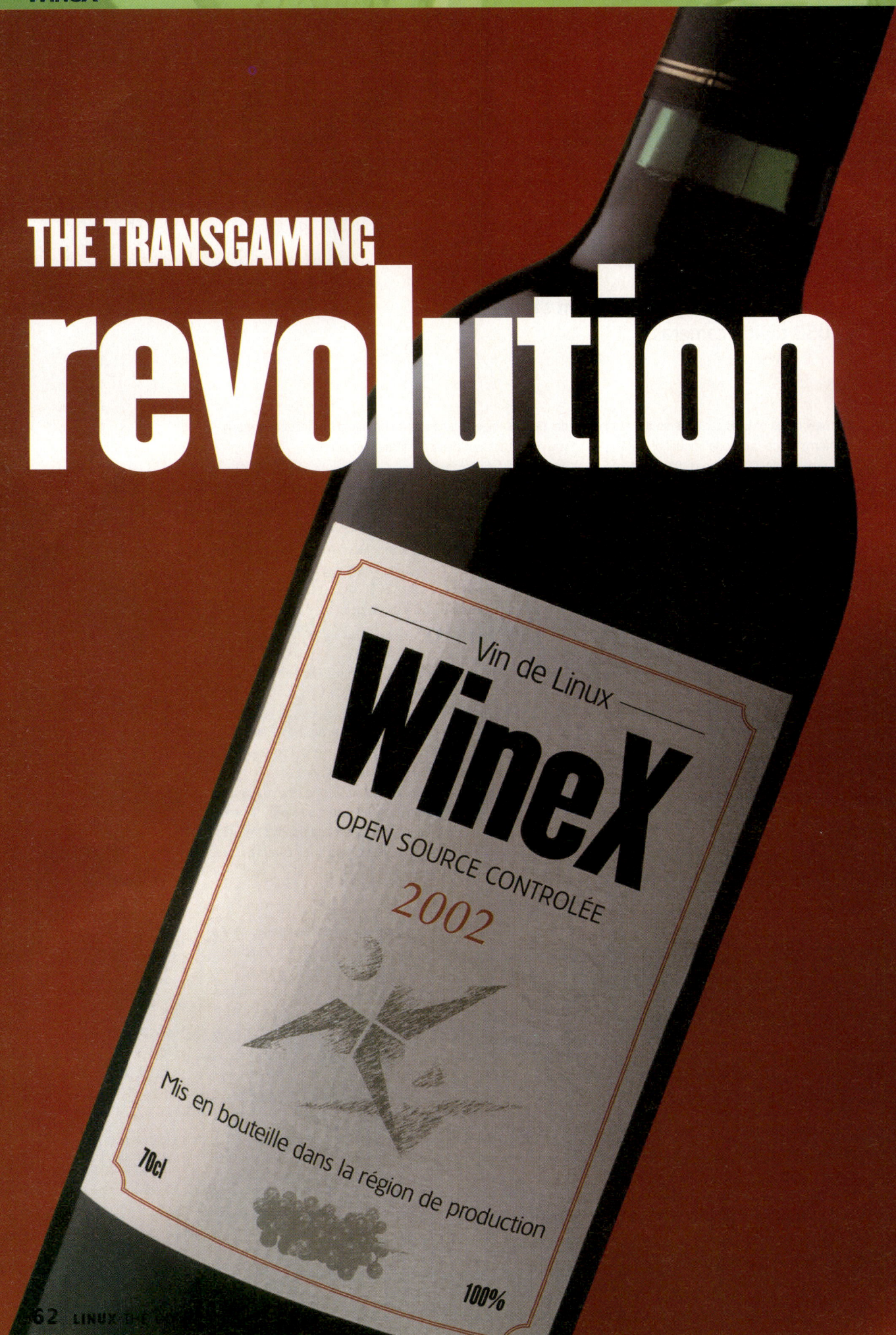
You can now happily use your camera, view the contents of the drive or copy all the files off it. One word of warning though. Although USB is designed to be hotpluggable, some of the USB device drivers for Linux are not 100% there yet. If you unplug the camera in the middle of copying or moving files, you can expect some data to get lost. Don't do it! Make sure any copying is completely finished before unplugging the camera! ■

Is my camera supported?

Because of the reluctance on the part of manufacturers to supply cameras, we can only say for certain that this works with the HP Photosmart 315 camera (which we just happened to have lying around the office).

It should, in theory, work for a whole range of cameras from different manufacturers, who support their devices in a similar way. If your camera is supported in Windows/Mac OS as a USB drive, chances are it should work under Linux.

THE TRANSGAMING revolution



Is Wine really the future of Linux games development? And what do Transgaming bring to the party? Nick Veitch has the answers.

Wine has been around for quite a while, and its development has seen many great leaps forward, and some not so progressive moments. It has had strong associations with Corel in the past, who leveraged the technology to launch a raft of tools (*Corel Draw*, *PhotoPaint*, *Word Perfect Office*) for Linux. Even Microsoft have taken notice of the technology, by adjusting their EULA document for Windows XP seemingly to try to legally halt its use. It has also stirred up ferocious debate amongst the Linux community. However, it also has some strong allies.

Wine started back in 1993 as a project to help Windows 3.1 applications run on Linux. It isn't an emulator, but rather a layer that sits between the program code and the underlying Linux system, which attempts to fulfill the needs of software making calls to Windows APIs. As it doesn't emulate the hardware or processor, you will need a Windows-compatible processor (*i.e.* x86) to run Windows software.

Wine is, in effect, a bunch of libraries which do the same for Linux as their counterparts on the Windows system. When the application makes an API call, *Wine* steps in to respond. Most of the issues with using software revolve around the user interface and system resources – for example, drawing a Window, accessing a disk, making a TCP request and so on.

For the most basic of applications, this is plenty of support, so your simple text viewers aren't going to be a problem. A lot of Microsoft software proves to be more problematic because, since they wrote the Windows API, they often use shortcuts and undocumented features which are very difficult to mimic in *Wine*.

Then there are problems caused by the fundamental differences between Windows and libraries – for example, Windows DLLs (dynamic linked libraries) have a preferred base address, and have a special fixup table when they need to be relocpated.

Running games

Games provide several greater challenges for the *Wine* project. In the first place, unlike most other software, games are usually very intensive applications that use various methods and shortcuts to 'take over' your hardware. Most of these are endorsed by Microsoft and some features of Windows are almost specifically oriented to games programming – *DirectX* for example. Then of course, games generally don't use the same sort of GUI interface as most windows apps, and often include their own highly customised interfaces, which rely on various different Windows resources. As these may not be the primary focus for the *Wine* project, they are supported to various levels of completeness at the moment. However, you can bet that any custom GUI uses the full extent of the capabilities available to it.

Before you even get to running the game, you have to install it though, and this is where *Wine* can fall down. There are many different types of installers, which all work in different ways. If you thought the installer was just a simple script you are much mistaken – Windows installers are very complex pieces of code that can easily require as much effort and skill to get working reliably as writing a small application. It's very annoying to know that an application works, but you can't install it because the install software doesn't. Another problem is copy protection, usually linked to the way CDs are mounted on the system.

Transgaming's focus for *WineX* is to develop the *Wine* technology in areas which have previously been overlooked or put off because they were not considered to be part of the core goals of *Wine*.

Non-gaming benefits

Gaming is probably one of the major factors behind hardware development on the x86 platform. The fastest, newest processors, graphics and sound cards are usually to be found in the hands of extreme gamers. Usually the

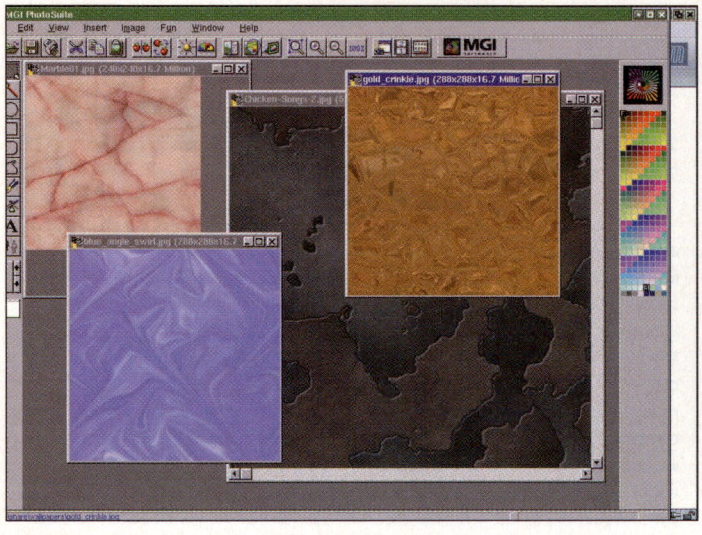
What is Codeweavers Wine then?

A vintage selection

The Codeweavers team have done a lot of work on the *Wine* project, and have also produced several wine-based technologies including *CrossOver Office* and *CrossOver plugin*. These last two are proprietary software that allow MS *Office* and Windows plugins to be run on the GNU/Linux OS.

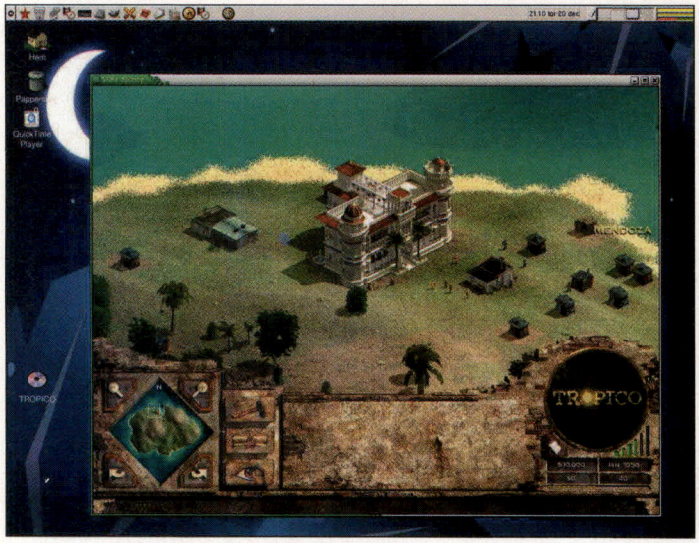
Codeweavers also produce a packaged version of *Wine*, which has a proper installer and configuration utility, and is much easier for the novice Linux user to set up and use. This version of *Wine* is straight from the main *Wine* development tree, but due to the time taken to package it up, etc, is not the very latest version of the software.

You can find out more about Codeweavers' contributions to *Wine* and their software at www.codeweavers.com



***DirectX* and installer technologies supported by *WineX* means more applications can run on Linux. Here's MGI *PhotoSuite*.**

***WineX* advances mean that even games which rely heavily on *Direct2D* are playable on Linux systems.**



A number of games, like *Tropico* shown here, will run but won't install properly on *WineX*. Yet...

Free Roaming Mode
FIRE for Next Target ALT-FIRE for Prev Target JUMP to Change Modes
Your text messages can only be seen by other Spectators

***Half-Life* and many of its popular modifications run on Linux, even for network play! Day of Defeat is well worth the download.**

games have caught up with the hardware in a few months and then everything moves on yet again. In the same way that games make extreme demands on the hardware, they also often make extreme demands on the underlying software too.

So if you can get a game to work, especially a cutting edge game, you can usually get other perhaps less exciting but perhaps more useful application to work.

One of the immediate benefits of *WineX* for application users is the support for many installers. Software which may previously have worked okay but would never install has now become available for use.

DirectX is also very important. Many graphics tools use *DirectX* to speed up 2D and (if appropriate) 3D rendering. Without *WineX*, they just wouldn't work.

Just an excuse?

Many detractors of *Wine* have complained that, far from helping out Linux users, *Wine* is actually giving people an excuse not to create Linux software. Company X can't be bothered to make a Linux version of their product if it may be possible to use the same code, saving themselves large amounts of money and development time.

Wine has been responsible for some important software ports to Linux though. Corel made a big thrust into the Linux marketplace when *Corel Draw* and *WordPerfect Office* were ported to Linux, something that was possible in a matter of months rather than years thanks to the *Wine* libraries. It may have been possible to create more efficient and Linux friendly code by re-writing everything from scratch, but it just wouldn't have happened. By making it possible to run this software under Linux, a market is created, which will encourage more native development – at least that's what the pro-Winers believe.

Whatever side of the *Wine* debate you are on, you must agree that *WineX* has made significant advances into cross-OS compatibility. Now if only I could get *Day of Defeat* to work through the company firewall, I'd really be sorted...

Licence changes

Wine changes mean changes to WineX too

Earlier in the year, changes were made to the licence for *Wine*. Originally developed under the very permissive X11 licence, the *Wine* developers have agreed to transfer further developments of *Wine* to the GNU Lesser General Public License (LGPL).

This has had some impact on Transgaming's strategy, as well as that of others who are co-operating with the *Wine* project.

In a briefing on the topic, Gavriel State, founder and CEO of Transgaming made the following statement:

"For TransGaming, specifically, there are a number of implications of the license change. First of all, *WineX* supports copy-protected software – almost every game available for Windows today relies on some form of copy-protection, and it is a necessary requirement for many of our partner companies. The LGPL would dictate that we publicly release the source code to our copy-protection support - an action which would violate the tenets of the US Digital Millennium Copyright Act (DMCA), as it would divulge critical knowledge about how such technologies work. Thus, revealing this information is not an option.

"The LGPL would also lock TransGaming into the long-term support of *Wine* even in a situation where our subscription objectives are not fulfilled. This imposes non-reciprocal benefit whereby the *Wine* project is furthered by TransGaming, but our corporate objectives and the needs of our users, partners, and customers are not.

"Given these considerations, we have chosen to adopt an alternate approach, architected by TransGaming's Ove Kaven, and another long-time *Wine* developer, Eric Pouech, who are collaborating with a good cross-section of the Wine team to begin the "ReWind" project (for details, see http://rewind.sourceforge.net).

***ReWind* is a continuation of the "X11" licensed version of the *Wine* tree and has received the endorsement of many *Wine* developers who are happy to have their changes applied to both *ReWind* and the LGPL'd *Wine* project."**

Transgaming WineX 2.01

- **DEVELOPER** Transgaming
- **WEB** www.transgaming.com
- **PRICE** subscription $5/month

Transgaming's *WineX* first really hit the headlines when released as part of a special bundle with the *Sims*. While popular, the *WineX* technology at that stage was far from reliable and bug free. At the time, the Sims was the only officially supported game and, although many other games were working to some degree, a lot of them were unplayable due to problems with interfaces or speed.

The *WineX 2.0* release catapulted the number of fully working, playable Windows games to over 80, and introduced support for a wide range of new features – most crucially, *DirectX 8*.

DirectX is the name for a group of technologies that comprises libs for 2D and 3D graphics, sound, music synthesis, in/out devices and even network connectivity. *WineX* handles both of the graphics elements very well, implementing *Direct3D* as a layer on top of *OpenGL*. You can run applications using *DirectX 8* under *WineX* using software rendering, but it isn't really practical – if you want to run a *Direct3D* game, you will need a hardware accelerated graphics card that is supported under Linux.

Subscribers can download the very latest packaged versions, which are available in a variety of formats, including rpm, deb and a binary tarball. There is no installer, but installing should be pretty straightforward, and there is now a user guide available for download.

WineX now supports several installer systems (though currently not the new MS Windows XP installer system), so usually you can run game installers from the CD, as you would under Windows.

As a bonus, *WineX* will map any start menu entries to your desktop menu, and map shortcut links to your desktop too. These will be set up to run *WineX* directly, so there is no need to open a shell to run your games – just click and go. *WineX* does do a great job of most installs, but check out the web forums and FAQs for advice on how to get stuff working. There is a wealth of information there which is very helpful, even if it just tells you your game won't work.

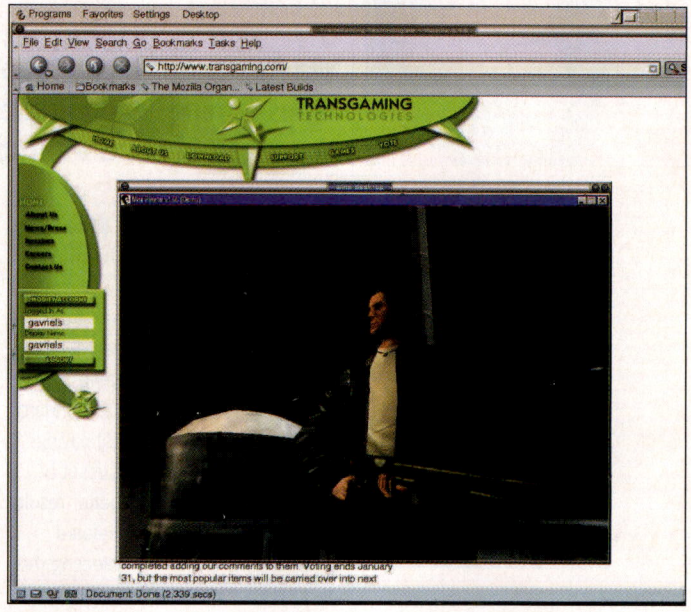
One of the advantages of subscribing is that you get to vote on what is worked on next. This has already meant that, for example, some games have been supported more fully, and certain aspects of *DirectX* have been speeded up.

Your *WineX* installation will create a .transgaming directory in your home directory, where config files and the Windows environment are to be stored. The *WineX* config is very similar to the standard *Wine* configuration file, but it might have been nice to include some sort of configuration tool for this. If you want to make all sorts of extra drives available to run *WineX*, you'll need to set them up yourself. With a bit of editing it's possible to use *WineX* with a Windows installation on the same machine, but be careful as it can be easy to confuse the two systems – *WineX* recreates any required elements of the Windows file structure itself. Having two registries can be confusing for the user. The best idea, if you want to do this, is to set up a specific user for it, so you can keep configs, etc, completely separate.

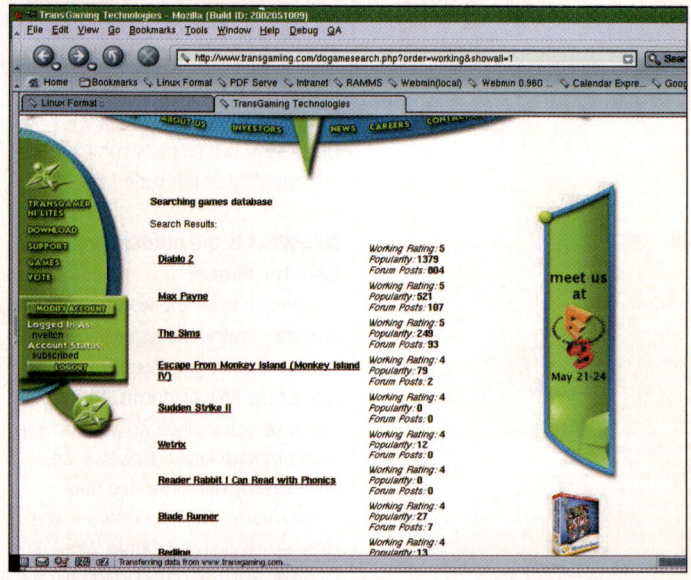
WineX has made what seemed like a forlorn hope not only possible, but easily achievable. A number of popular games are playable now, and the list continues to grow. The subscriber model is interesting (you can find out more info on this at the transgaming site) and perhaps brave, as you will get support for various types of problem. Perhaps the best thing about *WineX* is the contribution it has so far made to *Wine* itself, in terms of support for COM objects, installers and more.

Updates since 2.0

- **Better support for copy protection with SCSI CD-ROM devices, including ide-scsi used for most CD-Writers**
- **Support for CounterStrike 1.4 anti-cheat feature**
- **Fix for shell menu issues with KDE 3**
- **A fix for the incorrect error message "Can't find wine"**
- **A fix to better support some DRI cards in some Direct3D 8 games, such as *Max Payne***
- **Documentation now downloadable.**



***Max Payne* is just one of the flagship titles now running under *WineX*.**


TransGaming Technologies - Mozilla {Build ID: 2002051009}
File Edit View Go Bookmarks Tools Window Help Debug QA
http://www.transgaming.com/dogamesearch.php?order=working&showall=1
Home Bookmarks Linux Format PDF Serve Intranet RAMMS Webmin(local) Webmin 0.960 ... Calendar Expre...
Linux Format :: TransGaming Technologies
ABOUT US INVESTORS NEWS CAREERS CONTACT
Searching games database
Search Results:
Diablo 2
Max Payne
The Sims
Escape From Monkey Island (Monkey Island IV)
Sudden Strike II
Wetrix
Reader Rabbit I Can Read with Phonics
Blade Runner
meet us at
May 21-24
Transferring data from www.transgaming.com...

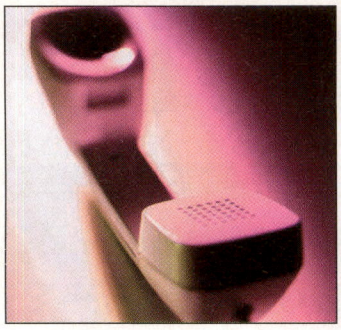
The game forums include helpful posts from other users, FAQs and a guide to how well games work.

VERDICT

Ease of use	8/10
Documentation	3/10
Features	9/10
Value for money	8/10

if you want to play that Windows game on Linux, subscribe now.

RATING 8/10



ON THE RECORD

We managed to catch up with Gavriel State, founder, CEO & CTO of Transgaming Technologies. Nick Veitch posed a few questions about WineX, Linux gaming and the future of Wine.

NICK VEITCH: Why was *WineX* formed as a fork from the main *Wine* development?

GAVRIEL STATE: The *WineX* tree contains substantial enhancements that are quite specific to games, whereas development in the standard *Wine* tree is less focused. By paying attention to this narrower subset of software, we can achieve better results without worrying about unrelated modifications to the tree affecting the performance of the programs our customers care about.

The *WineX* tree is also distinct from the WineHQ development tree due to differences in the licensing policy between the two trees – *WineX* is under the AFPL (Aladdin Free Public Licence), which restricts commercial redistribution of the code.

NV: What is the ultimate goal you have for *WineX*?

GS: Simply to be the world's quintessential seamless portability solution for bringing Windows games software to new platforms. This is a utopic objective since we are working with a moving target. However, we have already demonstrated how quickly TransGaming and *WineX* can keep up: when we debuted *WineX* in October 2001, we supported only six games. Now, just four months later, we support over 80 of the most popular Windows games.

NV: As well as games, the work done on *WineX* also helps support non-games software too (like *InstallShield* support), doesn't it?

GS: This is correct. In order to make *WineX* a user-friendly and complete solution, a great deal of work has been done on *InstallShield*, DCOM and other areas. This results from our desire to make *WineX* the simplest, most seamless and easiest-to-use portability layer available. Our objective is to serve our customers in the richest manner possible.

This means that *WineX* must be transparent so that anyone, technical and non-technical customers, can use it effectively. Some of our enhancement work also supports other business software applications.

NV: There has been some concern in the community about forking of the WINE project. Is there any intention to fold the code back into *Wine* in future?

GS: TransGaming already contributes large amounts of code back to the WineHQ tree. For example, we have contributed the majority of our 2D *DirectX* work, as well as substantial work on COM and OLE. We'll continue to contribute code in the future in accordance with *Wine's* own evolution.

NV: Given TransGaming's commitment to eventually make all of its work freely available, what will the company do with *Wine*?

GS: TransGaming is a strong *Wine* supporter and we are going to continue to work with the Wine community in a manner that makes the most sense for everyone. We are currently studying how best to move forward given the licence changes that have taken place in *Wine* recently.

NV: *WineX* has been criticised by some games developers as actually being a retrograde step – instead of encouraging developers to create original Linux games, it is almost giving them an excuse not to bother. How would you answer these claims?

GS: On the contrary, as most people know, the process of porting a game is very expensive, let alone developing games from scratch. If companies are going to invest the huge dollar amounts necessary to create a game, it makes the most business sense to pursue the largest markets.

While the consumer Linux market is growing rapidly, it has not yet reached the critical threshold needed to entice games companies to do any kind of solid development work in Linux on their own.

TransGaming is solving that fundamental problem; we are bringing as many popular games to Linux as fast as we can in an endeavour to increase the proliferation and utilisation of Linux on the desktop.

Once there is sufficient support and a demonstrable market, traditional game developers will start to add Linux to their portfolio of platforms to develop for. In the interim, Linux users want to play games and TransGaming is fulfilling that need.

NV: Other systems that try to co-operate with MS protocols, like *Samba*, for example, often find themselves hostages to development that they have no control over.

Isn't trying to create *DirectX* support, among others, for Linux just another case of a technology that will work for a while, but will then become broken by any subsequent releases?

GS: TransGaming has already demonstrated that we can keep pace with *DirectX* evolution. Again, we have over 80 games running on Linux in a completely seamless manner.

In December our subscribers told us that more than anything else, they wanted us to pursue *DirectX 8* support and compatibility. In less than 10 weeks, we had our first *DirectX 8* sample programs running, and shortly after we had one of the hottest games available, *Max Payne*, fully functional.

NV: TransGaming subscribers get to vote on where the development effort goes, whether on supporting specific games, or technologies. How is this system working out?

GS: Very well. In fact, our voters influence our development, selecting the titles and technologies they want TransGaming to work on, and we do our best to oblige them. *DirectX 8* and *Max Payne* exemplify this commitment to our subscribers.

NV: The voting system allows negative votes, but doesn't this mean that individual games always get negative results ?

GS: We allow negative votes in order for subscribers to express the difference between indifference to a title or technology and active dislike.

Our users know that every decision they make is about where our resources are directed. Some individual games receive negative votes, certainly – but that just makes it more evident which games our subscribers really want us to work on.

NV: The financial model for TransGaming seems to follow the 'Street Performer Protocol – the code is essentially free, but continued development depends on contributions. Is this working as expected so far?

GS: This model is working out well so far and our subscribers are very committed to TransGaming's continued success since we are working very hard to serve their needs. The continued evolution of the code certainly is contingent upon the continued support of our subscribers and all Linux users in general.

NV: Do you have other forms of investment to fund development, apart from subscribers?

GS: TransGaming is working on very unique technologies and applications that solve a strong need in a variety of different markets. As a result we have received strong support from the investment community who are committed to seeing us grow.

NV: How many subscribers do you have currently? I gather the target was 20,000.

GS: The target is 20,000 subscribers and we are very pleased with the tremendous support we have received from the gaming community. Although we cannot disclose the exact number quite yet, we are satisfied with the growth of the subscriber base in such a short time.

NV: How many active developers are contributing to *WineX*?

GS: TransGaming has assembled a truly outstanding team of developers who are also long time *Wine* contributors, in addition to the overwhelming support we receive from the *Wine* community. Our top-notch engineers come from all over the industry and possess a unique depth of Linux experience. Our team is over a dozen and growing rapidly.

NV: Is feedback from subscribers and users genuinely helpful?

GS: It's so helpful and important that we are incorporating a new subscriber element to our website very shortly as a gesture of appreciation to our existing members and an invitation for users to become subscribers. It will include some very entertaining and innovative contests, feedback opportunities, user profiles, prizes and more. Watch our website for up to the minute information.

This project is designed for more interaction with users and will feature some of their most inventive ideas and solutions. We're really excited about launching this process and to see what additional gems of wisdom and wit will come from it.

NV: To date there is only one 'officially' supported title, *The Sims*. Will there ever be any more?

GS: First, let me explain 'official' support. *The Sims* for Linux is a title that TransGaming has exclusive rights to and our *WineX* technology is integrated into the original source code enabling it to run on Linux.

It has been modified in several ways to make it more compatible with Linux users' needs: the sound system uses the free software Ogg Vorbis file format for playing music, the HTML export feature is compatible with Linux browsers like *Konqueror*, and video performance was carefully tuned for running with *WineX*.

We also have over 80 games, made exclusively for Windows, which run on Linux using our downloadable *WineX*, available to our subscribers. Many of these titles are still officially supported by TransGaming, in spite of not having licensing agreements with the original developers, and include such games as *Starcraft*, *Baldurs Gate 2*, *Sacrifice*, *Half-Life* and *Counter Strike*, among others.

These approaches, although overlapped in certain ways, are distinct. There will be other titles that we will be selling soon, where TransGaming is working directly with the original developers. We utilise both models because we want to gain a greater degree of presence and strength within the game development community while ensuring our users have an abundance of games to play.

"If companies are going to invest huge dollar amounts to create games, it makes sense to pursue the largest markets."

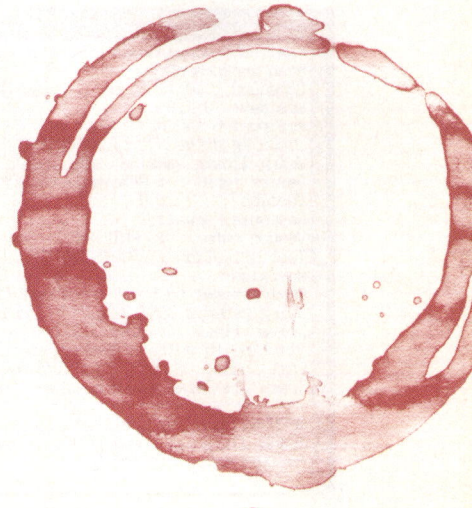
NV: Has there been any interest in supporting *WineX* from the commercial game developers?

GS: TransGaming is generating a lot of interest in the gaming world and we are currently in discussions with a number of companies regarding increased support of *WineX*. One of the primary reasons why TransGaming was able to obtain an exclusive licence to *The Sims* from EA was their interest in the strength of what *WineX* delivers.

NV: Some parts of *WineX* are licensed using the Aladdin Free Public Licence. Why did you choose this one?

GS: The AFPL strikes a great balance between our need to derive revenue from our products and the needs of users and developers who want to get their hands dirty with source code.

The AFPL restricts commercial redistribution of our code, but still allows the curious and more technically minded users to experiment with the code – quite a number of them have already been helpful to the *WineX* effort. ■



CD WRITING

Burning with Linux

PART 1 Robert Smith shows you how to burn CDs easily at the command line.

Storage on your computer is costing less and less each month. A decent 80GB drive only costs £80, meaning that space is no longer a premium. Transporting data also used to be a big problem as floppy disks only have a 1.44MB capacity and most people did not own a CD-writing drive. You can now pick up a fast CDRW drive for an IDE interface for less than £60, and the media for as little as 19p per CDR. This means that transporting, and even backing up, data can be a very cheap and quick. Using a CDRW drive under Linux is easy, very powerful and will allow you to burn CDs even while working with other applications.

Support for CDRW drives

Most of the CDRW drives available today can be used under Linux. If your drive is MMC standard compliant then generally it will work with Linux. This should be documented in the drive manual. If you can't find out if it is MMC compliant, then the easiest way to find out if it will work is to try it out by following the tutorial. We will cover both IDE and SCSI drives.

Getting ready to burn

Before you can burn a CD under Linux you will need to configure your kernel to support your CDRW device. If you are running a modern distribution such as Mandrake, SuSE or Red Hat then your device is probably already configured. To see if your CDRW device has been identified by your current configuration, use

```
cdrecord --scanbus
```

You should get an output similar to this:

```
Cdrecord 1.10 (i686-PC-linux-gnu) Copyright (C) 1995-2001 Jörg Schilling
Linux sg driver version: 3.1.22
Using libscg version 'schily-0.5'
scsibus0:
0,0,0     0) 'TOSHIBA ' 'DVD-ROM SD-M1212' '1R14' Removable CD-ROM
0,1,0     1) 'PHILIPS ' 'PCA382RW IDE     ' ' 1.4' Removable CD-ROM
0,2,0     2) *
0,3,0     3) *
0,4,0     4) *
0,5,0     5) *
0,6,0     6) *
0,7,0     7) *
```

In this case there are two devices viewable under the SCSI interface. The Philips drive is the CDRW and is connected by the IDE interface. If you get a blank output, try the following commands if you have a IDE device:

```
modprobe ide-scsi
cdrecord --scanbus
```

This command will only work if the kernel has been compiled with the *ide-scsi* (SCSI emulation support) module. Most base distribution kernels will have this module available to use from the box. If you have a SCSI drive then you will have to initiate the *scsi_mod* module and the *sg* module. Again, use the *modprobe* tool to activate these.

If none of the above works, you will have to recompile the kernel. Get the kernel source from **www.kernel.org** and compile it in the usual manner. If you have not done this before, read the docs which come with the kernel source code for some handy hints. The more heavily modified distributions such as Mandrake have lots of extra modules added to the kernel.

To make sure that your machine operates properly afterwards, use the kernel source rpm instead of custom source as this will make configuration easier. Make sure that the following parts of the kernel are either compiled into the kernel or separately as modules along with all other needed parts of the kernel:

- SCSI emulation support (this is necessary for IDE CDRW drives. If you want your normal CD drives to work with cdrecord easily then make sure that you don't compile the standard IDE/ATAPI CDROM support)
- SCSI support
- SCSI generic support
- SCSI CDROM support (this is needed if you have a separate CD drive that works on SCSI)

Once the kernel is compiled and your boot manager modified, reboot your machine using the new kernel.

Software Requirements

The software needed for this tutorial is

- cdrecord
- mkisofs
- Tools to compile a kernel
- cdparanoia

For the X-based part of the tutorial (page 72 on), you'll also need

- GnomeToaster
- X-CD-Roast

A terminal showing a completed *cdrecord* cycle.

```
rxvt
rsmith85@rsdev:/home$ cdrecord speed=2 dev=0,1,0 -data backup.iso -audio -pad baz_lerhman_sunscreen2.wav
Cdrecord 1.10 (i686-pc-linux-gnu) Copyright (C) 1995-2001 Jörg Schilling
scsidev: '0,1,0'
scsibus: 0 target: 1 lun: 0
Linux sg driver version: 3.1.22
Using libscg version 'schily-0.5'
Device type    : Removable CD-ROM
Version        : 0
Response Format: 1
Vendor_info    : 'PHILIPS '
Identifikation : 'PCA382RW IDE     '
Revision       : ' 1.4'
Device seems to be: Generic mmc CD-RW.
Using generic SCSI-3/mmc CD-R driver (mmc_cdr).
Driver flags   : SWABAUDIO
Starting to write CD/DVD at speed 2 in write mode for single session.
Last chance to quit, starting real write in 0 seconds. Operation starts.
Track 01: Total bytes read/written: 547913728/547913728 (267536 sectors).

WARNING: padding up to secsize.
Track 02: Total bytes read/written: 54172824/54173616 (23033 sectors).
rsmith85@rsdev:/home$
```

Your CD drives may not function as before after this change, especially if you did not compile the standard IDE CDROM part of the kernel. Use the **cdrecord --scanbus** command to check that your drive is being recognised by the software after any needed modules have been initiated.

If both your CDROM drives are recognised by this command then they will no longer be occupying the /dev/hdx block device. Instead they will be recognised under the /dev/scdx devices and you should change your /etc/fstab to point to these devices to keep your system running properly.

Where next?

By now you have set up your CDRW devices to work with Linux. This means that we can get down to the nitty gritty of burning CDs. This can be accomplished in two ways, either via the command line, or by using a graphical user interface under X, such as *GnomeToaster*. The command line method can be more powerful, as all the potential CD burning options are available to you instead of just the ones represented by the GUI.

By using an X application, though, you can reduce the steeper learning curve of the command line and this will definitely be more familiar to a Windows user. I recommend learning both ways and then choosing the right method for the right job.

On the command line

The first step in producing a CD via the command line is to produce an image of the CD that is being produced. This is done via the mkisofs program on the command line.

```
mkisofs -r -o <path to temp iso dir>/<iso name> <directory to put on cd>/
```

This will create an ISO with the **<directory to put on cd>** directory in, which is Rock Ridge compatible. The ISO will be found under the **<iso name>** file in the path specified. This ISO image can then be burnt using *cdrecord*, but more about that later.

```
mkisofs -J -o <path to temp iso dir>/<iso name>
<directory to put on cd>/
```

This will create a Joliet compatible CD which will be friendlier to all Windows users. To create a El Torito image use the following command. Make sure that the image size is exactly 2880KB or else the image will be rejected by the computer.

```
mkisofs -b -o <path to temp iso dir>/<iso name>
<directory to put on cd>/
```

If you are having problems creating a large ISO file, you will have to edit your **ulimit**. If you are root then this can temporarily be done by:

```
ulimit 10000000000
```

This will only last as long as the terminal session; after that the default will return. When you change the user as well, the limit defaults. To permanently change this (beware as this could cause a security hole), edit your /etc/security/limits.conf file. Change the line:

```
*        hard    fsize    <value>
```

to

```
*        hard    fsize    700000
```

and then login again to permanently change it.

Burning the image

To burn the data image to the CD you first need to find out the SCSI id of the drive.

```
cdrecord --scanbus
```

Find the three digit number beside the CDRW drive, and write it down. Then type in the following command:

```
cdrecord speed=<speed at which to burn> dev=<scsi id>
<path to image file>
```

Example 1

To back up the /root directory onto a CD, you should use the following commands.

```
mkisofs -r -o /rootimage.iso /root/
cdrecord speed=24 dev=0,1,0 /rootimage.iso
```

Example 2

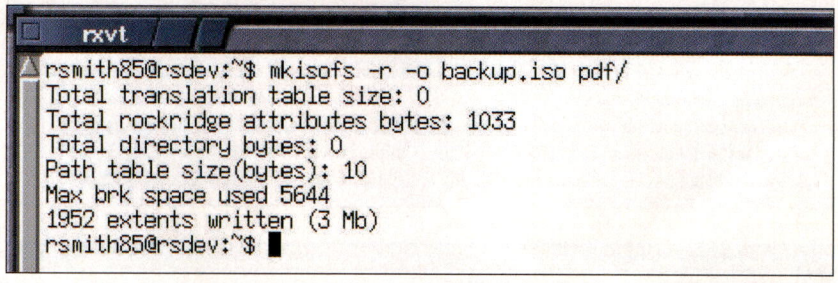
Backing up a single compressed archive onto a cd,

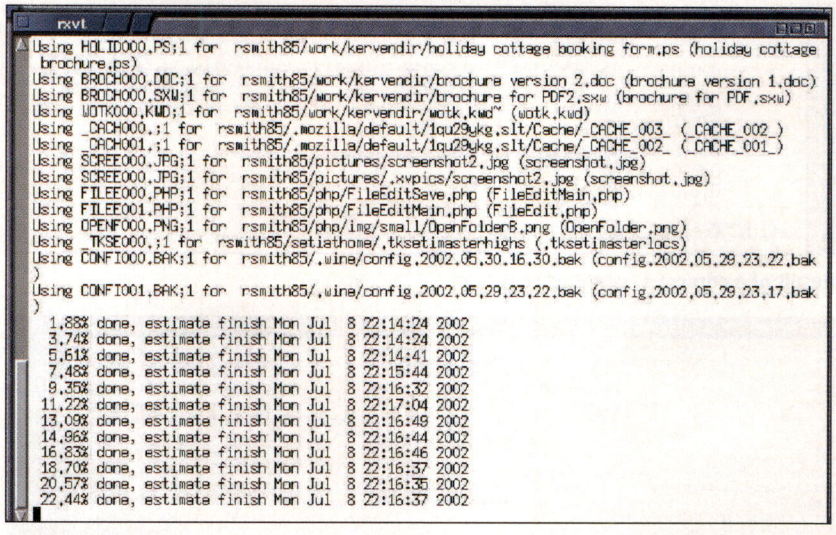
```
mkisofs -r -o /tmp/archive.iso /home/archive.tar.gz
cdrecord speed=24 dev=0,1,0 /tmp/archive.iso
```

Music onto a CD

To put tracks of music onto a CD you first have to get the tracks in wav format so that you can place them onto the disc.

Completed *mkisofs* command in a terminal.


rxvt
rsmith85@rsdev:~$ mkisofs -r -o backup.iso pdf/
Total translation table size: 0
Total rockridge attributes bytes: 1033
Total directory bytes: 0
Path table size(bytes): 10
Max brk space used 5644
1952 extents written (3 Mb)
rsmith85@rsdev:~$


rxvt
Using HOLID000.PS;1 for rsmith85/work/kervendir/holiday cottage booking form.ps (holiday cottage brochure.ps)
Using BROCH000.DOC;1 for rsmith85/work/kervendir/brochure version 2.doc (brochure version 1.doc)
Using BROCH000.SXW;1 for rsmith85/work/kervendir/brochure for PDF2.sxw (brochure for PDF.sxw)
Using WOTK000.KWD;1 for rsmith85/work/kervendir/wotk.kwd~ (wotk.kwd)
Using _CACH000.;1 for rsmith85/.mozilla/default/1qu29ykg.slt/Cache/_CACHE_003_ (_CACHE_002_)
Using _CACH001.;1 for rsmith85/.mozilla/default/1qu29ykg.slt/Cache/_CACHE_002_ (_CACHE_001_)
Using SCREE000.JPG;1 for rsmith85/pictures/screenshot2.jpg (screenshot.jpg)
Using SCREE000.JPG;1 for rsmith85/pictures/.xvpics/screenshot2.jpg (screenshot.jpg)
Using FILEE000.PHP;1 for rsmith85/php/FileEditSave.php (FileEditMain.php)
Using FILEE001.PHP;1 for rsmith85/php/FileEditMain.php (FileEdit.php)
Using OPENF000.PNG;1 for rsmith85/php/img/small/OpenFolder8.png (OpenFolder.png)
Using _TKSE000.;1 for rsmith85/setiathome/.tksetimasterhighs (.tksetimasterlocs)
Using CONFI000.BAK;1 for rsmith85/.wine/config.2002.05.30.16.30.bak (config.2002.05.29.23.22.bak)
Using CONFI001.BAK;1 for rsmith85/.wine/config.2002.05.29.23.22.bak (config.2002.05.29.23.17.bak)
1.88% done, estimate finish Mon Jul 8 22:14:24 2002
3.74% done, estimate finish Mon Jul 8 22:14:24 2002
5.61% done, estimate finish Mon Jul 8 22:14:41 2002
7.48% done, estimate finish Mon Jul 8 22:15:44 2002
9.35% done, estimate finish Mon Jul 8 22:16:32 2002
11.22% done, estimate finish Mon Jul 8 22:17:04 2002
13.09% done, estimate finish Mon Jul 8 22:16:49 2002
14.96% done, estimate finish Mon Jul 8 22:16:44 2002
16.83% done, estimate finish Mon Jul 8 22:16:46 2002
18.70% done, estimate finish Mon Jul 8 22:16:37 2002
20.57% done, estimate finish Mon Jul 8 22:16:35 2002
22.44% done, estimate finish Mon Jul 8 22:16:37 2002

***mkisofs* in progress.**

How CDs work

A many-layered wonder

CDR media is cheap to buy and so this is what most people are using. A CDR is generally made up of four layers. The following is a description of this from the bottom of a CD upwards.

■ Polycarbonate substrate layer
This is the see-through plastic that makes up the bulk of the CD.

■ Organic Dye
This is the clever part of the CDR. The layer of dye is changed by the laser in your CDRW to give pits and lands (data bits), which is how your information is stored.

■ Reflective Layer
Made out of 24k gold or silver-coloured alloy. This is what makes the bottom of a CD shiny.

■ Protective Layer
This protects the data from scratches.

When stored, the data is then read from the organic dye layer through the polycarbonate layer. Scratching the top surface causes more damage than the bottom surface because this removes the data, instead of just deflecting the laser. The information is then read using a different laser.

CD File Formats

The good thing about standards is that there are so many to choose from...

As with your hard disk, CDs can be formatted to hold different types of data. The following are the types of filesystem used on a CD.

REDBOOK – The standard for audio storage on a CD. This is pretty much wav format audio stored straight onto a CD, instead of into a file system. This allows the tracks to be read by hi-fi systems.

ISO-9660 – This is the standard file system type on a CD. Level 1 ISO-9660 gives the common DOS naming scheme of 8+3 characters. Level 2 improves on this, but makes the files less compatible with MS-DOS.

ROCK RIDGE – This is the Unix format of an ISO-9660 file system. It allows for longer names containing more characters, longer file names, symbolic links and mixed cases. It is only viewable properly by Unix-style systems, but can be read with ISO-9660 restrictions on non-compatible operating systems.

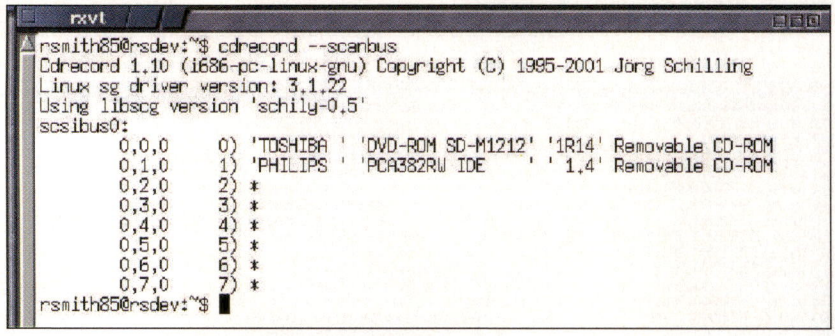
HFS – This is roughly thought of as being the Apple Macintosh equivalent of the ISO-9660 file system. It is incompatible with systems that do not have HFS support such as Microsoft operating systems, often giving the appearance of being faulty, but can be read by Macs, Linux and other operating systems with an HFS patch.

JOLIET – This is the Microsoft version of Rock Ridge. It has a file name limit of 64 characters but apart from that is pretty much an expanded version of the ISO-9660 file system.

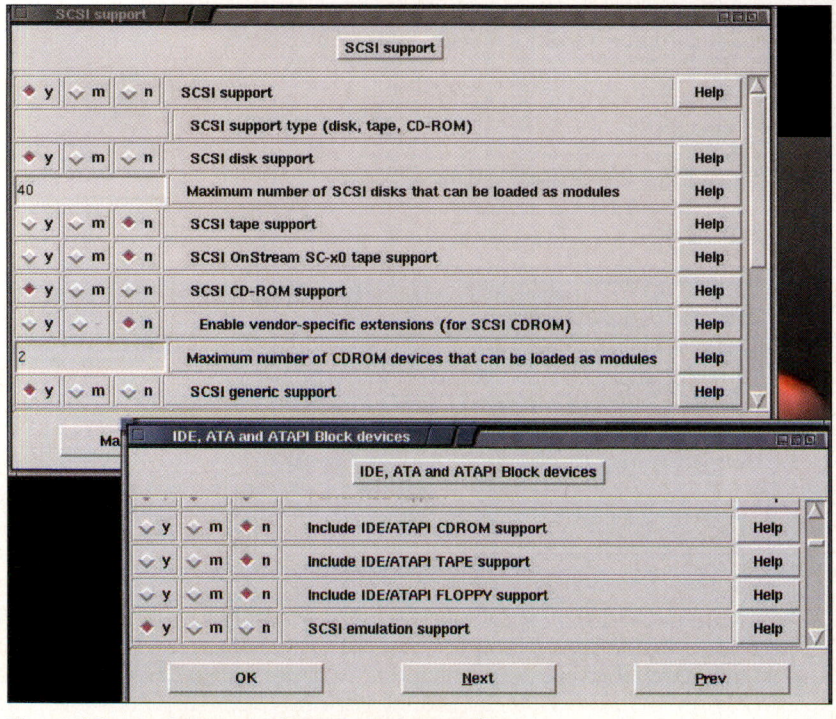
EL TORITO – A bootable file system that can be interpreted by a motherboard as a floppy image.

There are also some new formats coming onto the scene which are being intended to replace the ISO-9660 standard. One of these is called the ISO/IEC 13346 file system specification. This will enable more functions to be supported on CD and better multiplatform support.

Knowing about the different types of CD file systems available can be useful when producing a CD so you produce the most compatible CD for the intended audience. There is no point in using a Level 1 ISO-9660 file system if the CD is going to be used only on Unix compatible PCs.


rxvt
rsmith850@rsdev:~$ cdrecord --scanbus
Cdrecord 1.10 (i686-pc-linux-gnu) Copyright (C) 1995-2001 Jörg Schilling
Linux sg driver version: 3.1.22
Using libscg version 'schily-0.5'
scsibus0:
0,0,0 0) 'TOSHIBA ' 'DVD-ROM SD-M1212' '1R14' Removable CD-ROM
0,1,0 1) 'PHILIPS ' 'PCA382RW IDE ' ' 1.4' Removable CD-ROM
0,2,0 2) *
0,3,0 3) *
0,4,0 4) *
0,5,0 5) *
0,6,0 6) *
0,7,0 7) *
rsmith850@rsdev:~$

Result of a cdrecord --scanbus command.


SCSI support
SCSI support
y m n SCSI support Help
SCSI support type (disk, tape, CD-ROM)
y m n SCSI disk support Help
40 Maximum number of SCSI disks that can be loaded as modules Help
y m n SCSI tape support Help
y m n SCSI OnStream SC-x0 tape support Help
y m n SCSI CD-ROM support Help
y n Enable vendor-specific extensions (for SCSI CDROM) Help
2 Maximum number of CDROM devices that can be loaded as modules Help
y m n SCSI generic support Help
IDE, ATA and ATAPI Block devices
IDE, ATA and ATAPI Block devices
y m n Include IDE/ATAPI CDROM support Help
y m n Include IDE/ATAPI TAPE support Help
y m n Include IDE/ATAPI FLOPPY support Help
y m n SCSI emulation support Help
OK
Next
Prev

The modules needed to compile under make xconfig.

« Generally that will mean copying them (you must have legal permission to copy other people's music and files) off one CD and onto another. To do this via the command line, the easiest way is with a program called *cdparanoia*. The following command will copy all the music tracks off a CD onto your hard disk:

```
cdparanoia -wB
```

To do individual tracks, simply use the same command but specify a track number after the command.

This can then be burnt straight onto the CD in the same way as before with the audio flag, specifying the order of the tracks and separating the file names with a space:

```
cdrecord speed=<speed at which to burn> dev=<scsi id>
--audio <track> <track>
```

To mix data and audio tracks on one CD, simply type in the names of all the images and music tracks, separating them with spaces. If you are burning data tracks, use the -data flag before the image names. The same goes for audio with the -audio flag. This is so the cdrecord program knows what sort of data type it has to code.

Example 3

Burning hello.wav, seeme.wav and data.iso to a CD.

```
cdrecord speed=24 dev=0,1,0 -data data.iso
-audio hello.wav seeme.wav
```

CD copying using multi-sessions

When transporting files around, you will often find that you will only use a small amount of the total space on a CD. When done a lot this becomes wasteful, and although CDs are cheap these days, carrying one around instead of five, for the same amount of data, is a better option. This is when multi-session CDs are useful. A multi-session CD will allow you to add new tracks to a CD until it is full up. To do this you simply have to specify an extra flag while burning the disc. You use the following command to do this:

```
cdrecord speed=<speed at which to burn> dev=<scsi id>
-multi <path to image file>
```

Each time that you want to add a new track to this CD without finalising it, you need to use the **-multi** flag along with the flag for the data type being written (data or audio). On Sony CDRW drives another flag will have to be used. Look at the man page for *cdrecord* under **-multi**. The same may apply to other drives.

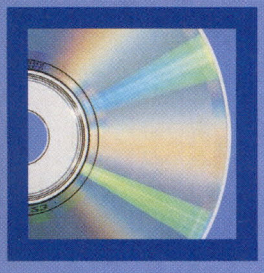
Multi-session CDs work in a different way from normal CDs. When a normal CD is burnt and finalised your CD is fixated in the drive. Fixation is when the Table of Contents (TOC), often called the lead in, is written into the small section at the start of a CD, and the lead out telling the CDROM device not to continue reading the CD for data is placed at the end of the CD.

With a multi-session CD, full fixation does not occur after every burning session. Instead, a new track is started, ready for fresh data to be added later, instead of the usual lead out. The table of contents is left unfinished. This kind of CD cannot generally be read by stand-alone CD players as they look at the TOC instead of searching for the first track. All the data on the tracks is linked together, making one large file system which is useful for prolonging the life of a CDR.

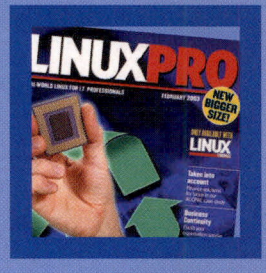
As you can see, burning CDs under Linux, even at the command line, can be very simple. Once you identify the best method for your own personal CD creation, be it Rock Ridge or Joliet, then creating more CDs becomes a easy task.

Over the next few pages we will look at burning CDs using two of the GUI packages which are available for X: *GnomeToaster* and the sturdy *X-CD-Roast*. ■

THE MAGAZINE THAT GIVES YOU MORE!



» Every issue comes with 2 CDs or a DVD packed with the latest software



» **FREE** 24-page Linux Pro magazine with every issue

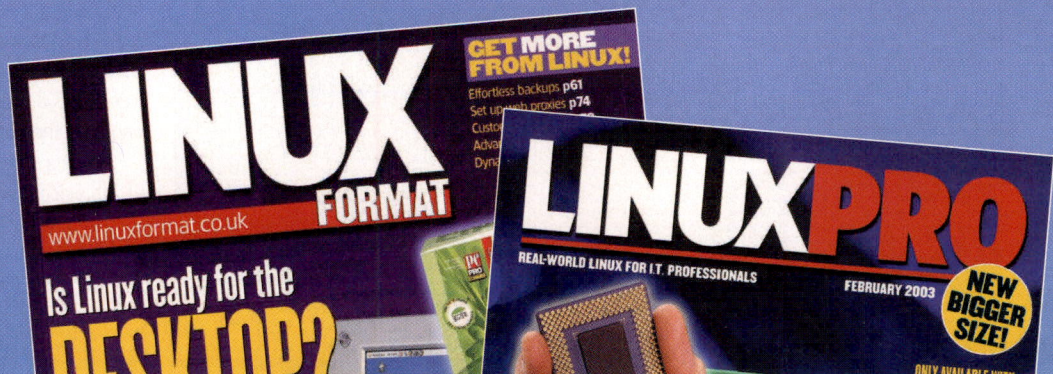
» Check out reviews of the latest linux hardware and software

» Practical and easy-to-use tutorials for all Linux users

LINUX FORMAT

THE UK'S BEST-SELLING LINUX MAGAZINE

Subscriptions Hotline
0870 444 8645
Please quote code LXFHAD



CD WRITING

Burning with Linux

PART 2 Robert Smith explains burning for the point'n'click generation, then delves into mixed mode CDs.

Now you know how easy and powerful it is to use Linux to burn a disc, we will look at the various options available for burning CDs under an X interface, and how to use both effectively. In this article we will look at using the *Gnome Toaster* and *KreateCD* software in particular.

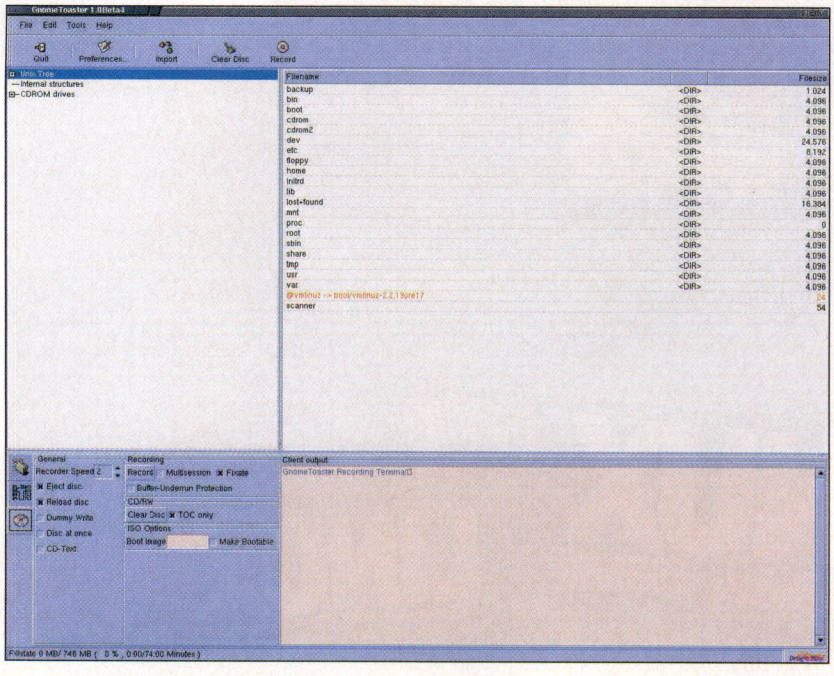
Gnome Toaster

Gnome Toaster is one of the better CD GUI burning tools under Linux. Although the interface may be confusing to begin with, if you have ever used *winoncd* in Windows, you should be able to master the program quickly and easily. If not, the following guide will get you started with the basics.

The interface is divided into three main parts, with a toolbar and menubar along the top. The left-hand panel (long and thin) allows you to access your filesystem, along with any CD devices or SCSI data devices you may own. The main large panel shows you the files and folders in a certain part of your filesystem, and the tracks on the other storage devices. The bottom panel provides you with your CD layout options, along with the main burning options to create your end CD.

The bottom panel is further subdivided via side graphic tabs to provide these options. Tab 1, which is shown by a folder symbol, is the ISO image creation part of the software. This is where you drag and drop files from your filesystem to produce a data CD. The next tab is shown by a T symbol, which stands for Track. This is where you add audio tracks and existing data tracks, then reorganise them so that they are in the order you require.

Burn your way through it, with the huge number of burn options in *Gnome Toaster*.



The third tab is indicated by a CD symbol. This panel opens up and gives you all the CD burning options that are available under *Gnome Toaster*.

Configuration

The majority of the settings are already configured when you start *Gnome Toaster* and unless you know exactly how the majority of the commands work, we wouldn't recommended you change most of the options. The main configuration options that you need to make sure are working correctly are the configuration of your CDROM and CDRW drives, and pre-caching for audio and data. To configure these options, click on the preferences button on the toolbar. This should open up a new window.

Configuring CDROM + CDRW devices – Click on the CDROM and Recorder Setup" tab and check to see that your devices have been recognised. If they have not you can add them using the ADD button at the bottom of the page, or choose for *Gnome Toaster* to scan your SCSI bus again. If you have a IDE CDROM you can use this by adding a new device and pointing all the options to its normal mount point and its /dev/hd* device instead of a /dev/scd* device file.

Make sure that it is not selected as a writer, as these need to be found by SCSI emulation to work. Click on your CDRW and then click the edit button. Make sure that the "This drive is a CD writer" option is checked.

Data Caching – To set up data caching you have to select the following options. Under the Common tab, click the "Precache Data tracks" option. Then click on the ISO9660 tab and select the "Precache before writing" tab. This will ensure that caching has been activated. Caching is useful as it means that your system requires less resources when it finally burns and there is less chance of a buffer underun.

Audio Caching – This is extremely useful when recording audio off multiple CDs as it records the audio data straight onto your PC. It also reduces the chance of errors – for example, if the audio CD has any physical problems such as scratches which stop consistent reading of the audio.

Creating a data CD with Toaster

This process is extremely simple and can be very quick and easy to do. Select the part of the file system that you wish to put on the CD using the left-hand side panel. Use the **+** and **-** signs next to the roots to expand and contract the trees respectively so that you can select the parts of the filesystem. Then click on the folder that you wish to burn and drag it into the Filename/Filesize window in the bottom panel of the program (set to the file tab).

The data should then be added to the ISO filesystem. If you wish to place the contents of the folder into the root of the filesystem, select the folder you want in the left-hand side panel, then select all the files and folders that appear in the right-hand table and drag them into the Filename/Filesize window. If you

only want to copy individual files, navigate to the folder which contains the file(s) using the left panel and then select and drag them in the right panel.

Adding audio

This is also a very simple process and can be achieved in several ways. The first is to drag the audio track directly from a CDROM drive. To do this, under the CDROM drives section of the left panel, use the **+** and **-** boxes to select the CDROM drive that contains the audio CD with the tracks on. Then drag the required audio tracks from the right-hand panel into the Bottom panel set to the Track tag.

Depending on the settings that you chose in the Configuring *Gnome Toaster* section above, the track will autocache and this will allow you to import tracks from other CDs. You can also create an audio CD by dragging either a WAV, OGG or MP3 file from the Unix tree part of the left panel into the Tracks section of the bottom panel. *Gnome Toaster* will automatically convert the files into a burnable format.

Multiformat CDs

This is just a matter of creating a data CD and then adding audio tracks afterwards. Unfortunately, one limitation of *Gnome Toaster* is that you have to put the data track as track 1 on a CD or it won't burn the disc. This problem will hopefully be sorted soon, but for the moment if you wish to burn a multiformat CD under *Gnome Toaster* when you put the CD into a audio cd player you will have to skip the first track.

Burning options

When you come to the burning stage of the process you can do this in two ways. You can click the icon in the toolbar labelled Record or you can try the more advanced option, using the interface provided in the third tab of the bottom panel. The Record option uses the default options that are configured as standard and should be used if you are happy with the standard options. If you need to specify any different parameters such as burn speed and multisession compatibility, the bottom panel should be used. On this panel you can select a few of the burning options available. These are explained below:

Eject Disk – If you want your computer to eject the CD when finished, select this.

Dummy Write – This is where the write process is completed with the writing laser turned off. This is useful when checking if the CD will burn correctly.

Multisession – This leaves the CD open so that more tracks can be added later, such as extra data.

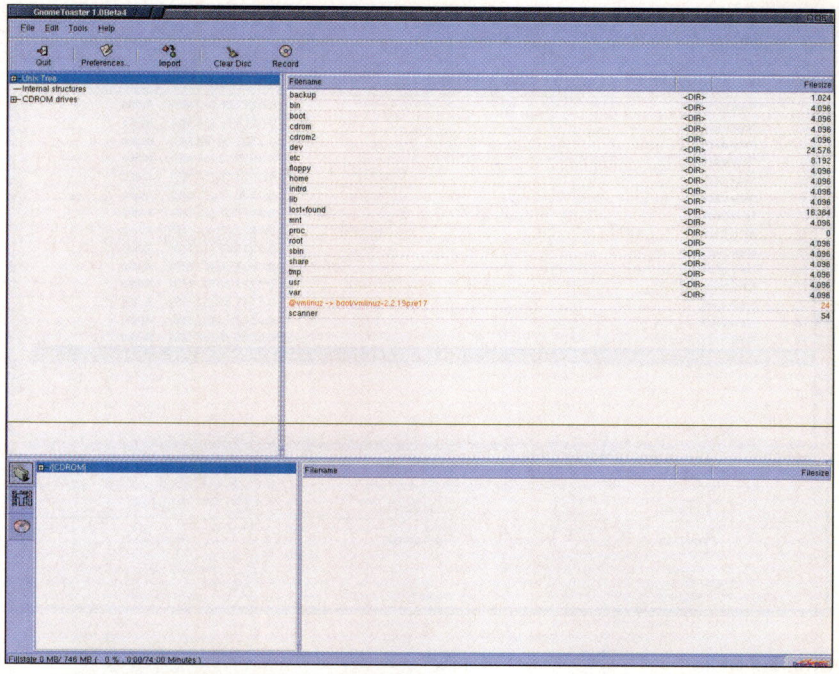
Fixate – This dictates if the CD should be closed so that no more tracks can be added to the CD and it can be played in an audio CD player.

Buffer Underrun Protection – If your CDRW has support for this, use it to protect your computer from information underrun where your computer doesn't send data fast enough.

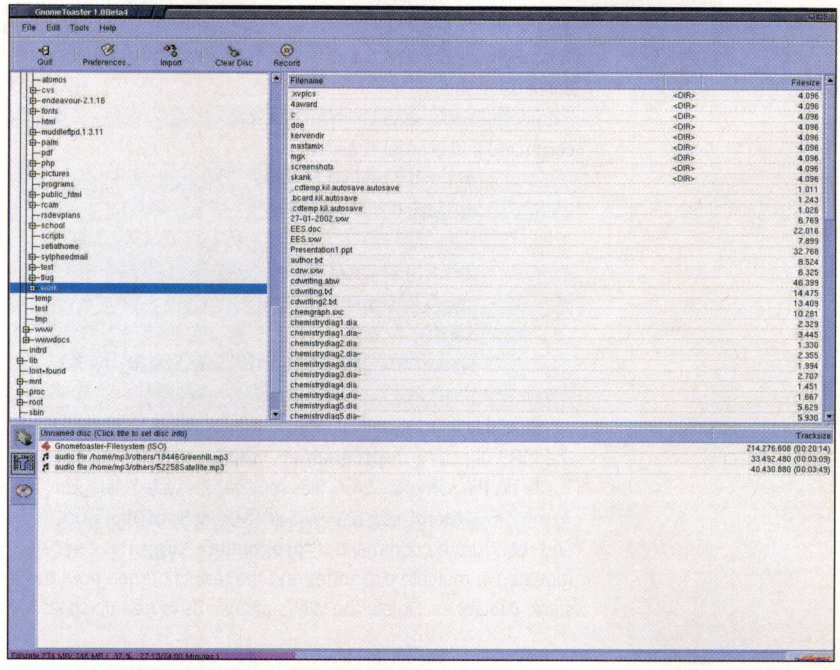
Clear Disk TOC only – When blanking a CDRW, this specifies if only the TOC is blanked.

The Bash way

Using X GUIs to burn an audio CD from MP3 and OGG files makes this an easy process, but it can also be accomplished via a simple *Bash* script or a Perl script. This way involves a for loop which converts all the files back to WAV files on the fly and burns



ISO editing made easy, with *Gnome Toaster*.



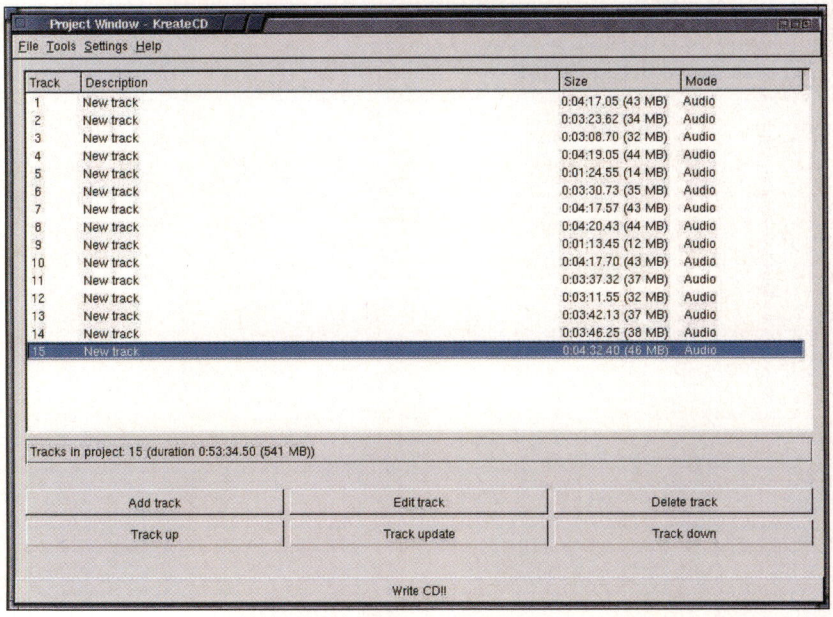
Adding data and audio tracks to a CD in *Gnome Toaster*.

them to CD. It then finalises the CD. The main problem with this is that the MP3s or OGGs have to be in the same directory and they are burnt in alphabetical or numerical order, rather than the order you specify.

The advantage of this method is that it uses the least disk space and is faster in the long run. This script should be run from the directory in which the MP3s are stored.

```
#!/bin/sh

for mp3 in *.mp3; do mpg123 --cdr - "$mp3" |
cdrecord -audio -pad -nofix
done
for ogg in *.ogg; do ogg123 --cdr - "$ogg" |
cdrecord -audio -pad -nofix
```


Project Window - KreateCD
File Tools Settings Help
Track
Description
Size
Mode
1 New track 0:04:17.05 (43 MB) Audio
2 New track 0:03:23.62 (34 MB) Audio
3 New track 0:03:08.70 (32 MB) Audio
4 New track 0:04:19.05 (44 MB) Audio
5 New track 0:01:24.55 (14 MB) Audio
6 New track 0:03:30.73 (35 MB) Audio
7 New track 0:04:17.57 (43 MB) Audio
8 New track 0:04:20.43 (44 MB) Audio
9 New track 0:01:13.45 (12 MB) Audio
10 New track 0:04:17.70 (43 MB) Audio
11 New track 0:03:37.32 (37 MB) Audio
12 New track 0:03:11.55 (32 MB) Audio
13 New track 0:03:42.13 (37 MB) Audio
14 New track 0:03:46.25 (38 MB) Audio
15 New track 0:04:32.40 (46 MB) Audio
Tracks in project: 15 (duration 0:53:34.50 (541 MB))
Add track
Edit track
Delete track
Track up
Track update
Track down
Write CD!!

Ordering audio around with *KreateCD*.

```
done
cdrecord -fix
```

The Perl way

This way will just convert the files to WAV files. These can then be burnt using the

```
cdrecord -audio <file 1> <file 2> etc.
```

command in the order required.

```
#!/usr/bin/perl -w
use strict;

foreach (@ARGV){
my @arr = split('\', $_);
$arr[0] =~ s/ /\\ /g;
system("mpg123 -w $arr[0].wav $arr[0].mp3");
}
```

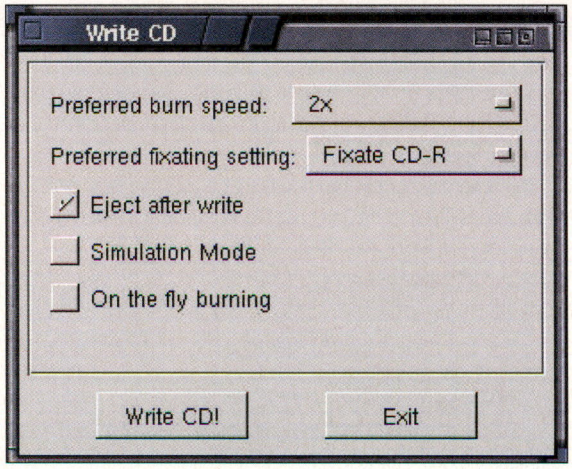
Save this in a file and run it with the following command in the MP3 directory. **./<progname> *.mp3**

To do the same for OGG files replace the system line with

```
system("ogg123 -w $arr[0].wav $arr[0].ogg");
```

and then run the command **./<progname> *.ogg** This can be repeated in multiple directories and the results burned from the same directories. Delete the WAVs afterwards to free up space.

Checking your options before the burn.



KreateCD

This is the best *Qt*-based KDE offering for CD burning. It has simple wizards and a great design interface, which means that it is very simple to use. When first started, a wizard interface comes up asking you what sort of CD you wish to burn. Choose the type of CD which you want to burn, or choose Classic mode. The Advanced mode is similar to that of *Gnome Toaster*, so if you are happy with this style use this.

When you first start *KreateCD* you have to configure the program once so that it knows how to burn a CD using your CDRW. To do this, click on the Settings menu and then the Configure KreateCD option. A new window will appear. Click on the SCSI tab and using the radial button, select which of your devices is the primary writing device and which is the primary reading device. To configure the speed at which your machine burns, simply click on the Burner tab and configure speed and post burn options. The ISO Defaults tab will allow you to configure the sort of ISO9660 filesystem that is created, such as a Rockridge compatible ISO file.

Audio CD mode

To add audio tracks using this interface, click the Add Track button. Then click on the Track Source drop-down menu. This will bring up three options: CD Track, File or Filesystem. To add a track from a CD, use the CD Track option and it will scan your reader device and give you a list of the tracks.

Select the one you want and then click ok to add the track. To add an MP3, WAV or OGG file to the audio CD, choose the File option, navigate to the file, select and then click OK twice to add it to the line up. Do this for all your files, then use the Track Up and Track Down buttons to order the tracks. When you have finished, click the Write CD button at the bottom. Change any final burning options and then click Write.

Data CD only Mode

Navigate through the tree on the left panel, and then drag across files and directories to the right panel. Click on the "Calculate ISO-image size" button to check how full the CD is. Once you are happy click the Write CD button at the bottom. Creating a data CD really is that easy.

CD Copy Mode

This is one of the best features of the package. Put the CD you want to copy in your CDROM device and a blank CD in the CDRW device. Click Write CD and the CD will be copied after you have chosen the final options.

Classic Mode

This allows you to easily create multimode CDs. To change the type of track that you are inserting, click on the Add Track button. Under the Track Type drop-down menu select Audio for an audio track or Mode1 for a Data CD.

If you choose Audio, follow the same instructions as above in the Audio CD tutorial to add the track. If you chose Mode 1 and selected filesystem, then a window such as the Data CD Only interface will come up. Use this to drag and drop the files into place. Click OK to save the track ready for burning. If you have a data track on a CD that you wish to copy onto a new CD, click the CDROM option under Track source, then select the data track on the CD after the TOC has been read.

Types of Burning

All at once, or a bit at a time

There are many ways in which your CD writing device can write the information to the CD, including Disc-at-Once, Track-at-Once and Packet writing.

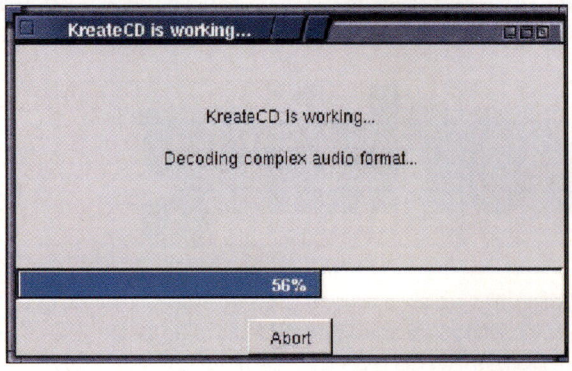
Disc-At-Once (DAO)
In this method the CDRW drive writes all the data to the CD without turning off the writing laser. This causes the disk to be finalised after it has been written. In this way, it requires a blank CDR and cannot support multisession.

The advantage of this method is that there is no gap in between the tracks burnt on the CD, in comparison to Track-at-Once which leaves a two-second gap between all of the tracks. With an audio CD this is the difference between the tracks appearing continuous or with large gaps in between.

Track-At-Once (TAO)
With this burning style the CDRW device's laser is turned off after each track has been written, even if more than one track has to be burned in a session. This gives more flexibility as you can use multisession CDs, partially used CDs and you can leave the CD open. Unfortunately, you get a two-second gap in between each track, which may not sound as good on an audio CD. Track-At-Once can be used with variable-gap. This means that you can control how large the gap is in between the tracks burnt. The gap can be lowered to as little as $^{2}/_{75}$ of a second.

Packet Writing (PW)
This is where you can format a CD much like a hard disk, and then write information onto it in small increments. This is very useful when copying small amounts of data from one computer and taking it elsewhere. Unfortunately, this too has drawbacks – the computer requires certain software to access the data and only one track is burnt to the CD, which is data.

All CDRW devices will support TAO mode, and the more modern ones will support TAO with variable-gap. Unfortunately, only a few CDRW devices support DAO, but this can be found out by experimenting with the settings in such programs such as *XCDRoast* and *Gnome Toaster*. Packet Writing is still in its infant stage under Linux. More information about this can be found at **http://packet-cd.sourceforge.net** – you will need an experimental kernel module to get this working. You will also have to enable UDF while building the kernel.



***KreateCD* takes away the hassle of burning complex audio files with its inbuilt decoders.**

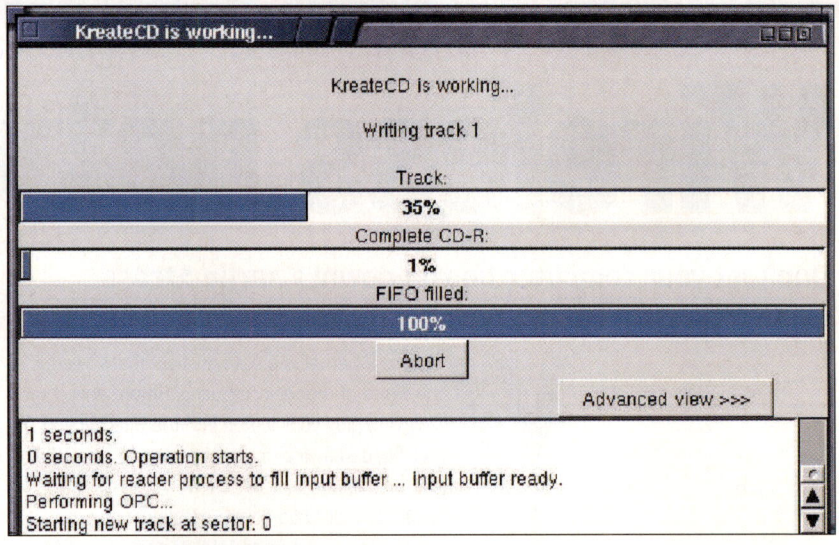
There is a large number of X-based CD authoring tools now available for Linux. If you are not happy with the way either *Gnome Toaster* or *KreateCD* work then try another – with dozens of programs appearing day by day there must be one that you can become acquainted with. Have a look for the following packages: *gcombust*, *cdrtoaster*, *cdbakeoven*, *XCDRoast*. Alternatively, you can always just stick to the command line.

Unfortunately, the *KreateCD* package is still very much in development. I personally found that the Copy CD option under the package didn't work correctly with audio CDs on my Linux setup, although it worked fine with data-only CDs. The same thing occurred when using the audio track from CD under the other modes.

User interface

Audio grabbing works from all other X-based CD writing packages, so this leaves me to believe it is a bug with the code in *KreateCD*. The method for adding tracks can be effective when working with small numbers, but when adding simular files this can become a bit time consuming. However, the great user interface provided by *KreateCD* more than makes up for the speed difference and for a beginner this is a good option.

The simplicity of *KreateCD* means that almost anyone can create CDs under Linux. In many ways the process is actually easier than using programs under Windows such as *Nero* and *Create-A-CD*. Coupled with the command line, you will find that authoring professional quality CD layouts can be easily accomplished. Creating a backup CD can be so quick that you have no excuse not to do so, and this will mean that your data will be well protected from any crashes or hardware failures that may occur.



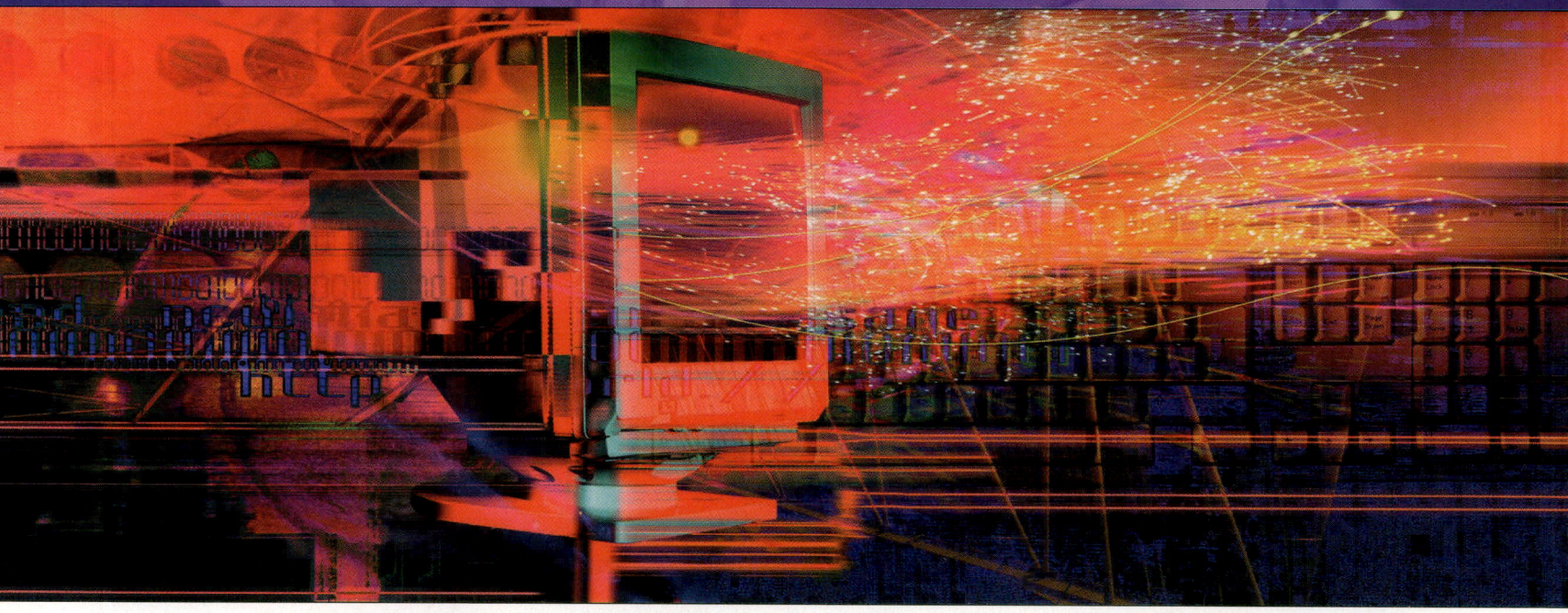
***KreateCD* happily burning a CD.**

KreateCD is easy and fast, with its drag and drop system creating an easily usable interface, although its stability may be questionable. Alternatively, *Gnome Toaster* has very powerful ISO layout tools and audio conversion tools. This choice will mean that no matter what you want to do, how you want to do it and how quickly, you will always be able to accomplish the task effectively under Linux. ■

Further Information

Useful links

http://gnometoaster.rulez.org
Gnome Toaster homepage
www.kreatecd.de
KreateCD homepage
www.xcdroast.org
XCDRoast home
www.fokus.gmd.de/research/cc/glone/employees/joerg.schilling/private/cdrecord.html
CDrecord homepage
www.tldp.org/HOWTO/CD-WritingHOWTO.html
CDwriting HOWTO
www.linuxlinks.com/Software/CD_Writing
CD writing software links
www.linuxformat.co.uk/modules.php?op=modload&name=phpBB_14&file=index&action=viewforum&forum=1&6794
The *Linux Format* magazine help forum (which is always useful)



WIRELESS ETHERNET WITH 802.11B

Wireless networking

Don't let your computer tie you down! Charlie Stross explains how you can free yourself from network cables.

Wireless Ethernet is a hot new field that's sweeping the world of mobile computing. Before now, if you wanted to plug your laptop into the Internet you either needed a wire or a mobile phone, and PPP dialup over mobile phone was *slow*.

802.11b is a standard for radio communications between computers, running at speeds of up to 11Mbits/sec – that looks like an Ethernet connection as far as the computers are concerned (see the box *Wireless Ethernet Explained* for details, and information about related standards). In this tutorial we'll look at hooking up a Linux laptop to a private Ethernet via 802.11b.

To use 802.11, you need two or more 802.11 cards, and possibly an access point. Most 802.11b cards are PCMCIA cards for laptops – the exceptions are usually PCI-bus (internal) PCMCIA adapters with a PCMCIA card already mounted! This is because desktop PCs don't really gain much from wireless networking. Such cards look like any other PCMCIA card, but for a small stubby aerial sticking out of the card slot. (Apple iBooks use different cards that use an aerial built into the laptop's case.)

An access point is an 802.11 base station, with antennae and an Ethernet port, and they serve the same purpose as an Ethernet hub, allowing multiple 802.11 client systems to talk to each other and the wired-up network. Note that you only need an access point when running a wireless network in certain modes, although this will certainly be the case if you want your laptop to talk to an existing LAN or fixed Internet connection. See below for details of the modes access points can run in.

There are drivers for a variety of 802.11 cards in the Linux kernel, as of version 2.4. The situation has been complicated a bit by some vendors who produced early cards that used competing, non-DSSS protocols, running at different frequencies. Only a few companies actually produce chipsets for 802.11 interfaces; most of the vendors are actually resellers who re-badge the equipment. You can find a list of the different types of chipset, and the companies that use them, at the following website: **www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/**.

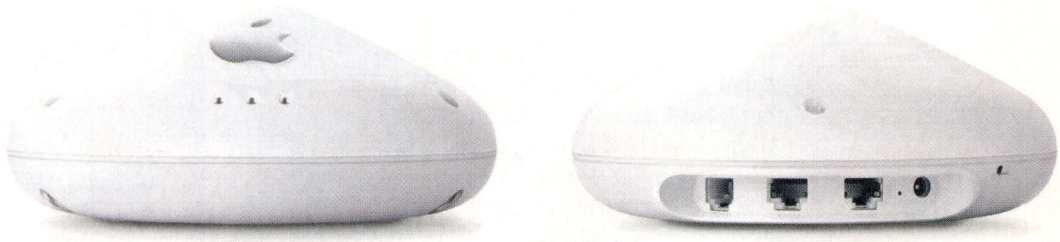
We would recommend that if you're buying an 802.11b card right now, you should try to pick one that uses the Orinoco chipset (produced by Lucent Technologies, sold by everyone, and formerly known as Wavelan). The reason for recommending this is that the ubiquitous Orinoco is well-supported under Linux – it's a bit like recommending a 3Com Ethernet card, back in the good old days. You can try a non-Wavelan card if you like, but you may find it slightly harder to configure.

It may not sound like the obvious choice for Linux, but an Apple AirPort makes a fine access point. The AirPort has a DHCP server, so that you can limit the 802.11b devices that access your LAN through it to known devices. It has a 10/100 Ethernet cable and a modem, so that in addition to using it to bridge wired and wireless networks, you can use it as a dialup router: in this configuration, your wireless machine can send packets out to the Internet via your ISP whenever you want a dialled connection. And although the configuration tool supplied by Apple runs on MacOS, there is a Java 1.2 configuration tool that runs on Linux at **http://edge.mcs.drexel.edu/GICL/people/sevy/airport/**.

The AirPort isn't incredibly cheap (street price is around £220, although it is falling towards $150 in the US), but if you don't mind some fiddling you can go cheaper. We tried out a NetGear ME102, which can be found for as little as £110 + VAT.

Like many access points, the ME102 is designed to be set up for the first time using a Windows utility connected via USB. Once assigned an IP address on the wired LAN, it can be managed by SNMP (simple network management protocol), but early versions of the firmware used by this device are insecure and leave it

vulnerable to a denial of service attack. While the AirPort behaves like a router, the cheaper access points are simple hubs that allow a bunch of wireless machines to talk to a wired LAN through them – they don't have DHCP servers or do dialup. On the plus side, if you're not afraid of risking prosecution, you can boost the output power on an ME102 to meltdown. See **www.wi2600.org/mediawhore/nf0/wireless/docs/802.11/WAP11/fun_with_the_wap11.txt**.



Apple Airport – it's shapely, fully-featured and, of course, works with Linux.

Setting up the laptop

The easiest way to get 802.11b working is to ensure that you're running a recent Linux distribution – with kernel 2.4, PCMCIA and the *wireless-tools* package installed. The main distributions come with the kernel wireless card support compiled as loadable modules – the *wireless-tools* package provides user-space tools for configuring the cards. Because almost all wireless systems are laptops, you should ensure that your system has PCMCIA support and then check out the FAQ at **www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/** to work out which kernel module you need to install to drive your card. PCMCIA support should be automatic on recent distributions such as RedHat 7.0 or later, and SuSE 7.0 or later.

If you need to build a custom kernel on an older distribution, you should consult the PCMCIA-HOWTO first, as you may need to add patches or modules to your kernel. Remember, the key items you need are: 802.11b drivers for your card, PCMCIA support and the *wireless-tools*, available in source code from **www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/**.

If you've got a card based on the Lucent Wavelan, insert it. The PCMCIA drivers should beep twice in a high tone if they've successfully loaded a recognised card, but you might need to explicitly install a kernel module. If so, in a root window, type:

```
insmod wvlan_cs
```

(or whichever the module name for your card is – *orinoco_cs* or *wavelan2_cs* are possibilities. Again, you will need to check the FAQ referenced above to work out which kernel module to use).

If you type **lsmod** (or **/sbin/lsmod** if it isn't in your path) you will see a listing containing something like this:

Module	Size	Used by
orinoco_cs	4832	1
orinoco	27760	0 [orinoco_cs]
hermes	3328	0 [orinoco_cs orinoco]
nfsd	64880	4 (autoclean)
ds	6832	2 [orinoco_cs]
i82365	23296	2
pcmcia_core	43008	0 [orinoco_cs ds i82365]
iptable_nat	12656	0 (autoclean) (unused)
ip_conntrack	12848	1 (autoclean) [iptable_nat]
iptable_filter	1728	0 (autoclean) (unused)
ip_tables	10496	4 [iptable_nat iptable_filter]
usbcore	47264	1 [uhci]

See how *orinoco_cs* depends on *pcmcia_core*, and *hermes* – the high-level interface to the 802.11b stack – depends on *orinoco*, which in turn sits on *orinoco_cs*.

If you type **dmesg** to see the kernel messages, you should also see something like this towards the end of the listing:

```
hermes.c: 12 Dec 2000 David Gibson
<hermes@gibson.dropbear.id.au>
```

»

Wireless Ethernet Explained

The tangle-free connection options

Let's start with a non-technical description: the IEEE 802.11 standard – also known as WiFi – describes a system for sending Ethernet packets over Ultra High Frequency radio. Unlike traditional 802.2 wired Ethernet, there are no wires. Like 802.2, it's a packet-based protocol. So basically you use it to hook networks of computers together, into LANs or WANs, without using wires.

Now let's get technical: 802.11 networks run on gigahertz-frequency wavelengths – microwave frequencies – using spread-spectrum and frequency hopping to enable a bunch of radio transceivers to talk to one another without interfering. Spread spectrum "spreads" a signal's power over a wider band of frequencies, sacrificing bandwidth in order to gain signal-to-noise performance. The spreading process makes the data signal much less susceptible to electrical noise than conventional radio modulation techniques. Frequency hopping takes the data signal and modulates it with a carrier signal that hops from frequency to frequency over a wide band of frequencies. This reduces interference because an interfering signal from a narrowband system will only affect the spread spectrum signal if both are transmitting at the same frequency at the same time. A frequency hopping radio will hop the carrier frequency over the 2.4 GHz frequency band between 2.400 GHz and 2.483 GHz.

Traditional Ethernet works by allowing any network card on the LAN to spit out a packet whenever it feels like it. If it's mangled by a collision with a packet from another computer, the protocol dictates that each machine "back off" (wait for a while) then try again. The Direct Sequence Spread Spectrum (DSSS) radios in your wireless Ethernet card do more or less the same thing, varying frequencies instead of time to avoid packet collisions. The result is that a bunch of radio transmitters can all transmit in roughly the same frequency band at the same time without drowning each other out, which is a vital prerequisite for anyone running a network.

The IEEE standard 802.11 specifies how DSSS data communications should work. The 802.11b sub-standard established requirements for a wireless LAN running at 11Mbps on DSSS radios using the 2.4GHz band; the newer 802.11g standard (coming later this year) should use 5.6GHz frequencies and boost the bit rate to 54Mbps. It's important to note that 802.11, even though it runs on the same 2.4GHz frequency as Bluetooth, is *not* the same as Bluetooth. Bluetooth is a 1.2Mbps wireless networking protocol that is limited to a range of about ten metres. However, before you write this off as poor, consider that Bluetooth devices are physically smaller than WiFi cards – they can be crammed into CF cards – and cost a lot less to manufacture. They also use so little power that you can run one off a PDA or cellphone without draining its batteries. A Netgear MA402 802.11b card, in contrast, can shorten the lifespan of a laptop's battery by 30%, sucking almost as much juice as the backlight.

Bluetooth is designed for the emerging field of PANs, or Personal Area Networks. A PAN might typically include your palmtop, laptop, printer, cellphone and other detachable gadgets, such as a keyboard. Its goal is to replace cables between peripherals that don't need to communicate at high speed. 802.11b, in contrast, is designed to replace wired Ethernets and connect high-bandwidth devices at long distances, so in a year or three you might have a PAN of Bluetooth-connected devices, linked to the Internet via an 802.11g-enabled device.

Bluetooth is supported by Linux – for example, IBM released their *BlueDrekar* protocol stack as Free Software in 2001 – but there are fewer Bluetooth devices out there at this time.

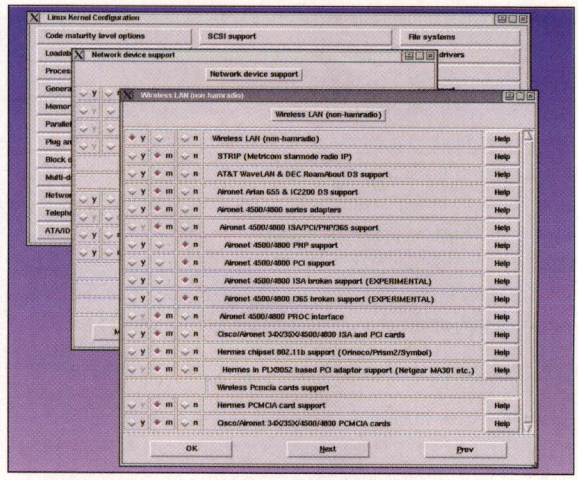
```
orinoco.c 0.07 (David Gibson
<hermes@gibson.dropbear.id.au> and others)
dldwd: David's Less Dodgy WaveLAN/IEEE Driver
orinoco_cs.c 0.06 (David Gibson
<hermes@gibson.dropbear.id.au> and others)
eth0: Firmware ID 1F vendor 0x3 (Samsung) version 0.08
eth0: Ad-hoc demo mode supported.
eth0: MAC address 00:30:AB:0E:86:17
eth0: Station name "Prism  I"
eth0: ready
eth0: index 0x01: Vcc 5.0, irq 3, io 0x0100-0x013f
```

What this tells you is that a card using the *orinoco_cs* low level driver has been detected and registered as Ethernet device eth0, with MAC address 00:30:AB:0E:86:17.

Having an Ethernet card in the system is all very well, but it doesn't do much good until you connect it to the network, which is where the fun starts. First you need to decide on a network type, then configure your access point, configure the 802.11b card to talk to the access point, then set up your routing tables. 802.11 provides for a number of different types of network, and of these, the most relevant to us are Ad-Hoc and Infrastructure. An Ad-Hoc network consists of a bunch of mobile devices with 802.11 interfaces, all talking to one another – there's no access point (hub) and no connectivity outside the local Ad-Hoc network. Ad-Hoc networks can be set up by simply assigning the mobile devices a channel (that is, a set of frequencies to operate in) and a network ID (or domain). In contrast, an Infrastructure network provides a designated access point which can forward packets from the mobile devices to an outside network, such as your wired LAN or the Internet.

WLAN cards operating on different frequencies don't communicate with each other. In addition, WLAN cards are assigned a different network ID (known in 802.11 jargon as an ESSID). Cards with different ESSIDs don't communicate either, so you should decide on a channel and ESSID and assign them to your access point, while keeping a note of the values for your mobile machine's card. The ESSID is a simple alphanumeric string that identifies all the people who want to be on the same network – it's not like an IP address, which must be obtained from RIPE, or an Ethernet card's MAC address, which is permanently burned in.

Wireless networking is notoriously insecure so if possible you should set up WEP encryption. Be advised that WEP relies on RC-4 and is notoriously weak – tools (such as *airsnort* and *ethereal*) exist that can break this level of encryption. If you're going to do anything significant over an 802.11 network, be prepared to use *ssh* a lot, or set up a virtual private network (VPN) on top of the wireless Ethernet. That said, it's probably best to get things working without WEP, then switch WEP on later.


Linux Kernel Configuration
Code maturity level options
SCSI support
File systems
Network device support
Wireless LAN (non-hamradio)
Wireless LAN (non-hamradio)
STRIP (Metricom starmode radio IP)
AT&T WaveLAN & DEC RoamAbout DS support
Aironet Arlan 655 & IC2200 DS support
Aironet 4500/4800 series adapters
Aironet 4500/4800 ISA/PCI/PNP/365 support
Aironet 4500/4800 PNP support
Aironet 4500/4800 PCI support
Aironet 4500/4800 ISA broken support (EXPERIMENTAL)
Aironet 4500/4800 I365 broken support (EXPERIMENTAL)
Aironet 4500/4800 PROC interface
Cisco/Aironet 340/350/4500/4800 ISA and PCI cards
Hermes chipset 802.11b support (Orinoco/Prism2/Symbol)
Hermes in PLX9052 based PCI adaptor support (Netgear MA301 etc.)
Wireless Pcmcia cards support
Hermes PCMCIA card support
Cisco/Aironet 340/350/4500/4800 PCMCIA cards
Help
OK
Next
Prev

Recent Linux distributions – and the modular nature of the kernel – may save you the trouble of a recompile.

Once you have set your access point to a given channel and ESSID, by whatever means come to hand (and this may mean running a Windows configuration tool to tweak your access point's configuration over a USB cable, which is outside the scope of this article!), you can turn your attention to your PCMCIA Wi-Fi card.

When the drivers are loaded, you configure your 802.11b card using the Linux wireless tools. These are included with most recent distributions, or the source code can be obtained as noted above. (Instructions for compiling them can be found in the INSTALL file in the source archive.) The wireless tools have a text-based interface and the programs are as follows:

- *iwconfig* manipulates the basic wireless networking parameters
- *iwlist* lists addresses, frequencies, bit rates and other items
- *iwspy* checks the link quality on a per-node basis
- *iwpriv* manipulates driver-specific (private) Wireless Extensions.

The most important of these programs is *iwconfig*. Just as *ifconfig* manipulates networking interfaces in the Linux kernel, *iwconfig* manipulates wireless networking devices in the kernel. And we want to bring up a wireless interface.

Suppose you've configured your access point to use the ESSID "private-wlan" on channel 1. (Essentially the ESSID acts like a callsign or network address, and channel 1 specifies the frequency the WLAN is going to run on.) You need to tell your Wavelan card (which the kernel recognises as eth0 – see the kernel messages earlier) to use these parameters. You can do so

802.11b and the Law

Make sure that your transmissions are legal

802.11b networking runs on the 2.4GHz waveband. This is reserved for scientific, medical and industrial (SMI) applications. It isn't licensed, but there are strict legal limits on how strong transmitters may be, and there is a blanket ban on commercial use of this wavelength. In California many cafés now offer 802.11b access points with broadband Internet connections for their customers, but a coffee shop offering this service in the UK would be breaking the law.

Also in California – and rapidly spreading – there's a movement for users with broadband connections to open up their access points to passers-by; see http://consume.net/ for details of this open Internet movement, which ultimately aims to immerse us all in a sea of peer-to-peer wireless networking, making conventional ISPs obsolete.

While some brave amateurs are experimenting with this in the UK, commercial use for any purpose is effectively illegal – you can use it within your office or business, but not for talking to customers or selling services, or even helping to sell a service.

However, hope is at hand. The Radiocommunications Agency (which handles spectrum allocation) has recognised demand for 2.4GHz spectrum, and for the 5.6GHz spectrum required for 802.11g. 5.6GHz is expected to be opened up for commercial applications by summer, and a public consultation on opening up 2.4GHz ended recently.

You can check out the review summary for yourself – it can be found at the following website: www.radio.gov.uk/ smag/24ghzinx.htm.

Companies can now obtain experimentation licences on a case-by-case basis (expiring after six months) to test possible commercial applications on the 2.4GHz frequency, and it looks as if commercial 802.11b services may be legalised soon.

by running the following command as root:

```
/sbin/iwconfig eth0 essid "private-wlan" channel 1
```

Remember, if you're using a different ESSID, or a different channel number, change the values accordingly. *iwconfig* has a slew of different configurable items – type **man iwconfig** to see the man page and a list of its options. You can use *iwconfig* to set a WEP key to use, and if you have set a key on the access point and enabled WEP encryption, this will be used to encrypt traffic between your card and the base station. Note that the WEP implementation in the stock kernel modules for the *orinoco_cs* driver and the ME102 access point are sufficiently broken that this may not work! If you get into trouble setting up WEP using the standard drivers, you may want to update to a newer kernel or try downloading the Linux *WLAN-NG* experimental drivers from **ftp://ftp.linux-wlan.org/pub/linux-wlan-ng/**.

You can also use *iwconfig* to enable power-saving modes on your card, if it's supported. This is particularly important due to the high power consumption of most Wi-Fi cards, which is often comparable to the power drain from the backlight of a laptop's LCD display.

If you've successfully set eth0 to the same ESSID and channel as the access point, you can now set up routing using **/sbin/route**, as you would for a normal Ethernet card. For example, if you are using the TCP/IP reserved network 192.168.1, and have a router at 192.168.1.1 while your laptop is 192.168.1.2, you can do something like this:

```
/sbin/ifconfig eth0 192.168.1.2
/sbin/route add 192.168.1.1 eth0
/sbin/route add default gw 192.168.1.1
```

This sets the TCP/IP address to use on eth0, adds a route via eth0 to the router and specifies that everything goes to the router (as a gateway) by default.

Some tools are slowly becoming available on Linux that put a prettier face on all this. While the distributions don't yet have graphical interfaces for configuring Wi-Fi cards, they can be expected to arrive very soon as wireless networking takes off. In the meantime, if you use *KDE* you may want to check out *Korinoco*, from **http://korinoco.sourceforge.net/**. This is a *KDE* utility for configuring your 802.11b card. Again, if you use an Apple AirPort access point then you are in luck because the java utility (referenced above) gives you a comfortable interface for configuring it.

Limitations of wireless Ethernet

The zeroth limitation of 802.11 is security. Bluntly, wireless Ethernet relies on radio transmissions. This means that your precious data is being transmitted where any snoop with a receiver and aerial may be able to pick it up. This is in stark contrast to conventional Ethernet cables, where you have a reasonable expectation of privacy over a private LAN. (See the box, *Wireless Ethernet Security*, for what to do about this.)

In conjunction with the legality issue (see the box, *802.11b and the Law*), the security problems place limits on what you can safely do with 802.11b networks. The protocol itself also imposes some restrictions on what you can do with a WLAN. The first and most important drawback is that microwaves (such as the 2.4GHz radio waves that 802.11b uses) don't go round corners or penetrate solid surfaces. This may actually be a good thing, depending on your point of view – if you are worried about drive-by hacking, knowing that there are stone walls thick enough to block the signal between you and any crackers can be very reassuring. Because access points cost significant amounts of money, you will want to site your access points carefully to give maximum coverage. One possibility is to place them in corridors connecting those rooms where you're going to use wireless devices – the signal is capable of carrying through a wooden door without much loss of strength, and a single access point in a stairwell is sufficient to provide coverage for most houses. Another possibility is to use a booster aerial, but be advised that in the UK there are legal limits on the strength of transmissions permitted on the 2.4GHz band, and you may be facing a hefty fine if your neighbours complain.

iwconfig usefully displays the local signal strength:

```
osh:/home/charlie # iwconfig
lo        no wireless extensions.

eth0      IEEE 802.11-DS  ESSID:"private-net"
Nickname:"foobar"
          Mode:Managed  Frequency:2.412GHz
Access Point: 00:30:AB:0D:XX:XX
          Bit Rate=11Mb/s   Tx-Power=15 dBm   Sensitivity:1/3
          Retry min limit:8   RTS thr:off   Fragment thr:off
          Power Management:off
          Link Quality:91/92  Signal level:-11 dBm
Noise level:-102 dBm
          Rx invalid nwid:0  invalid crypt:0  invalid misc:0
```

See the "Link Quality" and "Signal level" headings. You can use your laptop to map a small workspace for dead spots by simply carrying it around and typing **iwconfig** (or running *Korinoco*, which displays signal strength among other things).

If you're setting up a bigger WLAN, you may need multiple access points configured in Infrastructure mode, each acting as gateways to the same wired LAN. You can find pointers to how to configure this in the compendium of wireless HOWTOs at **www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html#whowtos**. For mapping signal strength on a large scale, you can set up the wireless tools and a suitable PCMCIA card on a handheld running Linux and *ethereal* (the Compaq iPaq has been used for this purpose), or use a special purpose tool such as the Locust (see **www.bvsystems.com/Products/WLAN/Locust/locust.htm**), designed for mapping signal strength and recording it against GPS location. ■

Wireless Ethernet Security

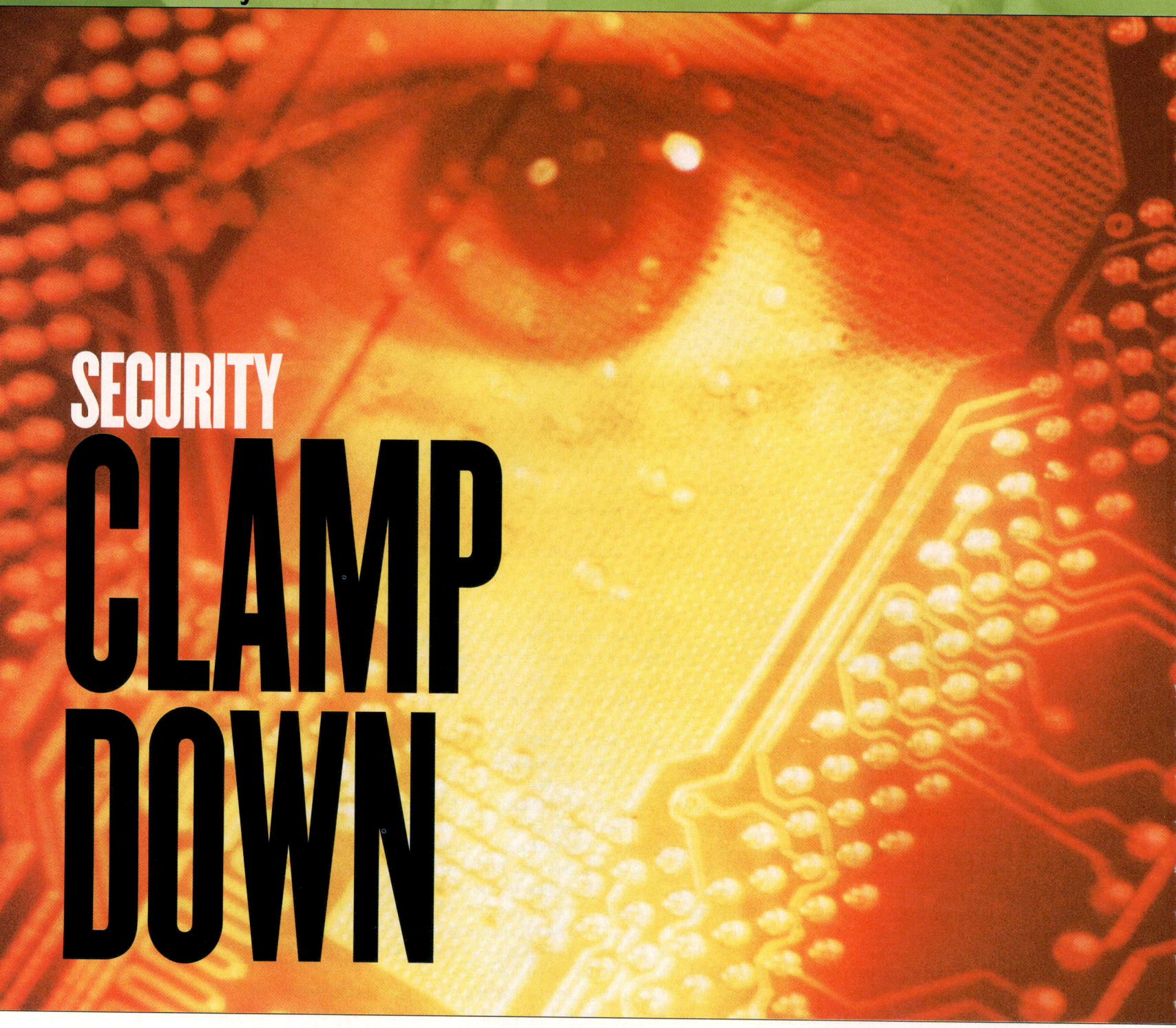
How not to send your data to your competitors

Anyone with a DSSS card can monitor traffic on an unencrypted 802.11b network. Because it's radio, it's effectively public. In order to secure the wireless network, a standard was adopted – WEP (Wired Equivalent Privacy). WEP uses the RC4 stream cipher to encrypt packets, using a 60-bit key. Unfortunately, the first 24 bits of the key are fixed, leaving just a 36-bit space to search. Weaknesses in the cipher algorithm are described at www.eyetap.org/~rguerra/toronto2001/rc4_ksaproc.pdf and *airsnort*, a tool that can passively monitor a WEP-encrypted network and extract the keys after monitoring a few hundred thousand packets, can be found at the following site: www.be-secure.com/airsnort.html.

Just in case you didn't get the message loud and clear: even using WEP encryption is no guarantee that your wireless network is secure! Newer systems (notably Apple's AirPort 2.0 access point and some new 802.11b and 802.11g cards) use 128-bit keys. These are much stronger and probably constitute an effective privacy-protection measure, but if you use a current-generation 802.11b network, you should treat it as under attack at all times.

This means keeping your access point outside your firewall, firewalling connections from your access point going out to the Internet at large as well as into your private network (unless you want your broadband connection to be targeted by drive-by spammers), and using *ssh* or IPsec to create a VPN for your confidential data transfers or login sessions.

Unless your entire WLAN is buried in a concrete bunker, of course...



SECURITY CLAMP DOWN

Mark Newby explains why network-level security is rapidly becoming more important, and what protective measures we can take.

Networking and security, the primary Unix strengths, are the main reasons that many choose GNU/Linux. If you don't have a network at home, you should take into consideration your connection to the biggest and most insecure network of them all: the Internet.

It's getting worse...

Not that long ago, Internet security was really a subject with which only IT departments were concerned but, unfortunately, network security now affects just about all of us in one way or another. Admittedly, it wasn't that many years ago that I too used to be pretty ignorant of the threats I faced when dialing up to the Internet with my home PC, even when running Microsoft Windows. Security was something other people had to consider – or at least that's what I told myself. Many people still think along these lines and Linux, for all its merits, can quite easily lull its users into a false sense of security through no real fault of its own.

Many of us have heard through the press and other media that Internet security is becoming more of a problem and that threats are multiplying. No longer is protection the sole concern of the IT departments of large businesses.

However, we're not always given an explanation of why this is so, which

does nothing to persuade people to change their approach to security.

What seems to be happening is that the number of security incidents on the Internet is increasing at a similar rate to the increase in the overall size of the Internet. **Figure 1** shows a plot of the increasing size of the Internet in terms of the number of connected hosts, or machines (ISC, 2002). **Figure 2** shows how the number of security incidents (reported to CERT/CC – the CERT Co-ordination Centre) appears to be following the expansion of the Internet with a slight delay (CERT/CC, 2002).

We probably all knew that the Internet was expanding, but the implications of this growth are not always quite as obvious. So as the threat of a successful attack on our computers from the Internet increases, so should our efforts to mitigate them.

This article aims to inform you of some of the dangers facing your computer – and what you can do to protect your data.

Home users targeted

Unfortunately, home users are considered to be an easy target by crackers, as they are considered to be far less switched on to the threats of security than businesses generally are. In my experience, Linux users seem to be more switched on about these issues than their Windows/Mac contemporaries, although, understandably, there is still usually a deficit in understanding compared to the IT professional in a business.

I often hear people claim that their home PCs would be of no interest to crackers, which just isn't true and possibly never was. "What would a cracker want with my pictures of Dorset coast and MP3s of Bob Marley?" Probably nothing. However, there are cracker groups who are very keen on using other people's computers to carry out their malicious activities, often very discreetly. This way, any crimes which are committed appear to originate from your computer and not theirs.

These compromises often go unnoticed by the owners of the computers for long periods of time as the crackers are able to hide or disguise all indications that malicious software has ever been installed or is currently running. In fact, some attackers claim that they are simply using spare capacity, thereby not harming the computer's owner (Zwicky, D E, *et al.*, 2000, pp5). Unsurprisingly, this is not the view that's usually held by the owners of the equipment!

Permanent connections are, of course, more attractive to crackers than those of an intermittent nature, such as dial-up modems, as crackers like to return on further occasions and don't want to find that the IP address they used yesterday is not there today. To make matters worse, 'always-on' connections such as ADSL are becoming more common, thereby providing more targets for crackers. How many advertisements for broadband services such as ADSL have you seen that also stress the increased importance of security with these services? None, I'd imagine, as it isn't much of a selling point. The threat is real, though.

But how can these crackers identify home users out of the huge collection of computers connected? It's far easier than you may think. IP addresses are normally assigned to organisations such as ISPs in blocks or ranges, called netblocks. It is a fairly straightforward exercise to find out, for instance, what netblocks NTL is assigned because the information is publicly available.

Although not all of an ISP's netblocks would be allocated to home users, we can assume the appropriate netblocks have been identified by now within the cracker community. The RIPE Network Co-ordination Centre (RIPE NCC) is responsible for assigning IP addresses in Europe and Figure 3 shows how easy it is to obtain the netblocks owned by ISPs – NTL in this case – from its website, found at **www.ripe.net/perl/whois**.

Keeping out crackers

Knowing that security is of rapidly increasing importance, we can now take a look at the various approaches we can take in terms of securing our computers and networks. For the home user, there's the question of whether a separate machine to act as a dedicated firewall is really the best way to go. For larger scale scenarios there's the question of network topology, benefits of a DMZ (De-

»

Mixing business and security

Justifying security efforts in the business environment

Unfortunately, effort expended on security is often hard to justify at a managerial or budgetary level; for much the same reasons that the product of software development is intangible until some application can be shown to be working. For instance, it often isn't until an organisation has their webserver, or internal network cracked into that people start to question whether they spend enough effort on security measures. Closing the stable door after the horse has bolted is, unfortunately, very common.

From a corporate point of view, an IT/IS department's budget is often expected to be reduced year on year. This poses a real problem because, as we've seen, the threat of a successful security attack is rapidly increasing, therefore, so should a business's efforts to protect against them.

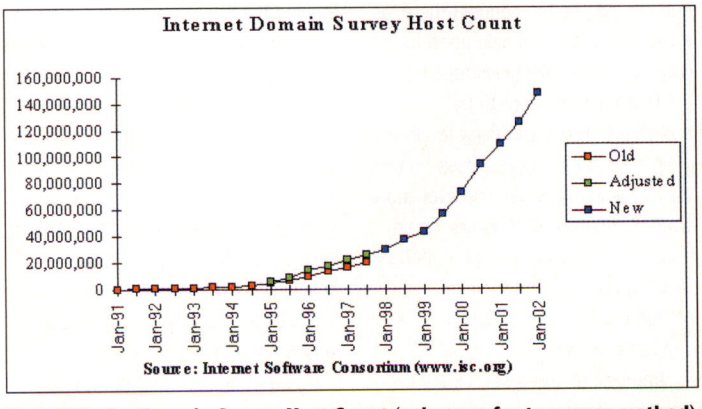


Fig 1: **ISC's Net Domain Survey Host Count (colours refer to survey method).**

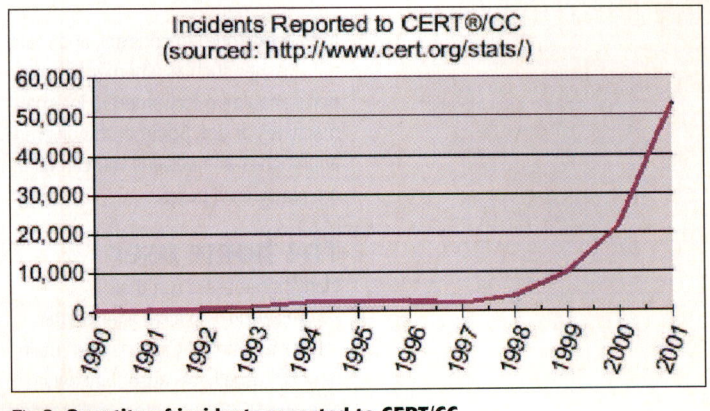


Fig 2: **Quantity of incidents reported to CERT/CC.**

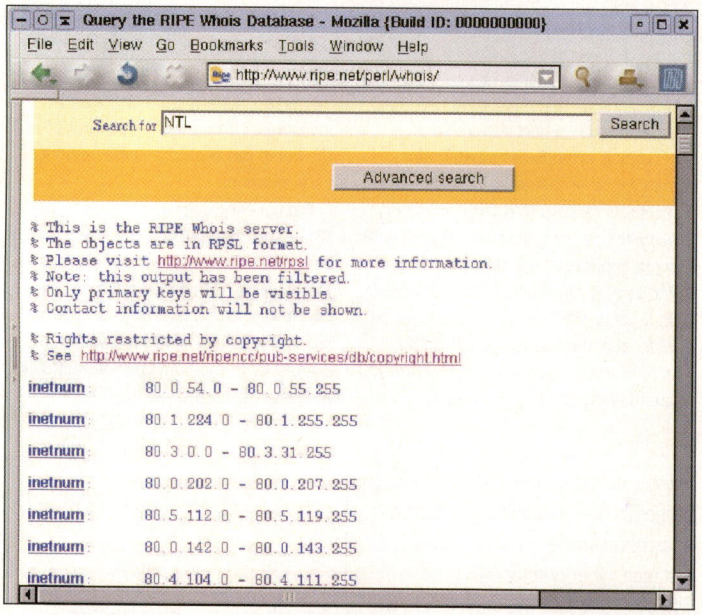

Query the RIPE Whois Database - Mozilla {Build ID: 0000000000}
File Edit View Go Bookmarks Tools Window Help
http://www.ripe.net/perl/whois/
Search for NTL
Search
Advanced search
% This is the RIPE Whois server.
% The objects are in RPSL format.
% Please visit http://www.ripe.net/rpsl for more information.
% Note: this output has been filtered.
% Only primary keys will be visible
% Contact information will not be shown.
% Rights restricted by copyright.
% See http://www.ripe.net/ripencc/pub-services/db/copyright.html
inetnum: 80.0.54.0 - 80.0.55.255
inetnum: 80.1.224.0 - 80.1.255.255
inetnum: 80.3.0.0 - 80.3.31.255
inetnum: 80.0.202.0 - 80.0.207.255
inetnum: 80.5.112.0 - 80.5.119.255
inetnum: 80.0.142.0 - 80.0.143.255
inetnum: 80.4.104.0 - 80.4.111.255

Fig 3: **Discovery of home users' IP addresses by way of an ISP's netblocks.**

"Crackers are very keen on using other people's computers to carry out their malicious activities – very discreetly."

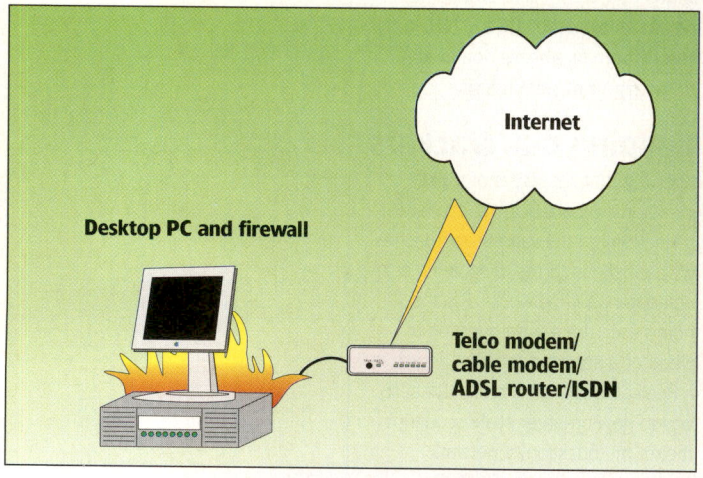


Fig 4: **Dial-up host acting as firewall and desktop/workstation.**

Militarized Zone) to consider, and many other issues such as whether certain ports should be forwarded to internal machines or not. Some of the different approaches and designs you can take are considered below.

The home user

Many home users run their *iptables/ipchains* packet filter on the same machine that acts as their main desktop, usually because they only own one PC. This topology is illustrated in **Figure 4**. Of increasing popularity, however, especially among us Linux users who like to dabble, fiddle and tweak on as many machines as possible, is to make use of an old PC for a dial-up firewall/gateway and to keep the desktop PC or workstation completely separate.

This is a very good idea, especially now that cheap second hand PCs running at around 100MHz can be picked up for £50 or less.

Dedicated firewall

The most important reason for having a firewall on a separate machine is for containment. For instance, consider the possibility that the firewall box becomes compromised. To what has the cracker now got access? If the firewall only does packet filtering, then the answer is not very much at all, but if the firewall is your main desktop PC with lots of 'spare capacity' and sensitive information like stored passwords, then the cracker has immediate access to this.

Personally, I also like the idea that my main desktop PC does not have to spend its time keeping the bad guys out, thereby keeping this capacity spare for when I really need it, like for intensive multitasking.

This general approach to containment and segregation is often applied to other services too – for example, keeping a webserver and email server on separate physical machines – although in terms of home users, this is hardly economical.

Figure 5 shows such a topology, where a simple **cross-over cable** is used to connect the two machines. If more than one machine needs protecting, like in my own home office where I have additional machines running different OSs, an inexpensive hub can be used to connect them, as a cross-over cable is only good for connecting two machines together.

There always seem to be businesses who are wishing to get rid of old PCs that just can't keep up with the demands of newer and ever more bloated versions of Windows. As an example, I managed to get a Pentium 133MHz for £25 from a local company, which I now use as a dedicated firewall.

Remember, these secondary machines don't need a monitor as they can be easily controlled over a network connection using *SSH* (*Secure Shell*) or telnet, for example. When there are only two machines, say a firewall and desktop, there isn't even any need for network equipment such as hubs or switches, since a simple cross-over cable will suffice (assuming Ethernet over twisted pair cable).

It is in this context that I'd like to mention Smoothwall, a slimmed-down Linux firewall distribution. It takes just a few minutes to install and configure in a few simple steps. There's no problem installing it on a resource-limited PC to turn it into a rock solid firewall, with **bastion host** characteristics. A network card in both machines and a cross-over cable is all it would need for the two-machine setup discussed above. Smoothwall can be downloaded for free under the GNU GPL (see the *Resources* box). A kind IT department may be able to create a short cross-over cable for you and NICs can be had for under £30.

Smoothwall enables the machine to act as the dial-up host, so the modem, ADSL route or ISDN card is connected to this machine, and a web-based interface can be used from your desktop in order to instruct it to connect as required.

DeMilitarized Zone

A business often needs to provide external access from the Internet (for example, remote staff or clients) to one or more machines and, with broadband connections such as ADSL, this is becoming more common with home users too.

An example is where a company wishes to host its own webserver to enable finer control or reduced costs. It would be extremely risky to run such a webserver inside the private network behind the firewall (for instance, using port forwarding) and, instead, a DMZ is normally used.

If the webserver was inside the firewall on the same network segment as servers storing critical data, or less well secured desktops and the webserver is cracked into, the cracker then has direct access to these machines and their data.

A DMZ is all about containment and is based around the 'what if...' method of analysis and design. Depending upon the exact details of

the topology, the use of a DMZ with a firewall is referred to as the *screened subnet architecture*. Any machines that require external access (from the Internet), like a webserver or email server, are placed inside a separate network segment known as the DMZ, which is segregated from all other network segments.

The firewall acts as a kind of three-way valve, providing interconnecting routes between the Internet, DMZ and internal network (where the PCs, printers and other such hardware devices are located). **Figure 6** illustrates such a topology.

In terms of the firewall's design, it is said to "not trust the DMZ", which normally equates to a policy of denying by default traffic originating from the DMZ and destined for the internal, private network segment in case of a compromise (except where the traffic is in response to a legitimate request for a connection from a client on the internal segment; for example, from a PC accessing the email server – packet filters such as *iptables/ipchains* can determine this).

Crackers often carry out network sniffing (listening to network traffic) from a compromised machine in order to try to pick out user name and password transmissions – say, from Windows Networking (SMB/NetBIOS), or POP3 traffic, which both transmit this data as plain, unencrypted text.

If the compromised webserver is on the same network segment as machines using these insecure protocols, it makes it really easy for the cracker to sniff out user names and passwords to enable him/her to log in legitimately in future.

The threat of network sniffing is a very good reason to locate an organisation's main email server on a firewall-protected internal network segment. A forwarding only email server on the publicly available DMZ receives all incoming email and forwards it to the internal server. This avoids having user accounts on the publicly available email server.

However, there are other methods of preventing user names and passwords traversing the network in plain text; for example, by using *ssh* instead of *telnet*, or tunnelling POP/IMAP through SSL, which encrypts all transmitted data.

Firewall config

There doesn't seem to be an agreed technical definition of a firewall, and it often represents not just a single machine running packet filtering, but it can do. The term originates from the building construction industry, where special-purpose interior walls are designed to stop the spread of fire from one part of a building to the next (Zwicky, E. D., *et al.*, 2000, pp21). It's one of those terms that gradually became well used in a number of different implementation schemes.

When talking strictly about Internet firewalls, they have been described as a way in which you can restrict access between the Internet and your internal network, which is a good summary.

So, we need a firewall, but we're now faced with a number of options. That is, do we:

- Use the integrated firewall services which are provided by our distribution (say, configured during installation)
- Develop our own iptables/ipchains firewall script manually to implement our filtering rules
- Develop our firewall script with a program that helps abstract the details to a higher level, *e.g.* with a GUI
- Use a dedicated firewall software package or an integrated distribution such as Smoothwall?

Choosing which option is the best for you depends on a number of different subjective issues:

- How much free time are you going to have available?
- Do you have, or indeed want, to spend your money on buying a product that comes with some form of ongoing support?

This is related to the first question of available time, as it otherwise takes effort to keep up with security announcements and software updates.

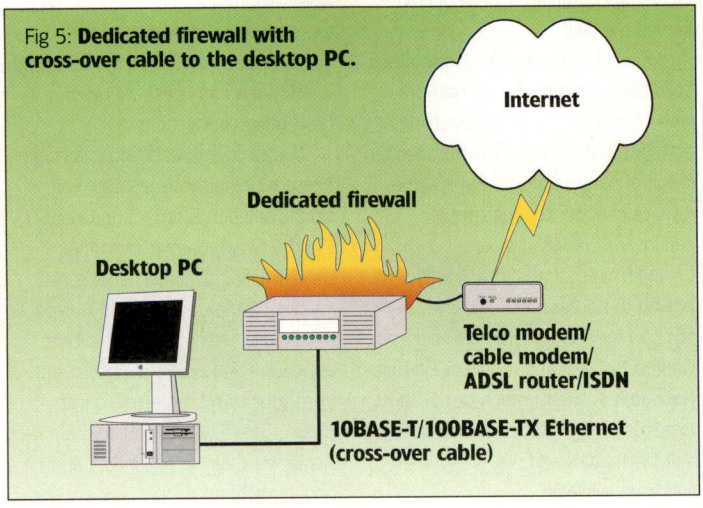


Fig 5: **Dedicated firewall with cross-over cable to the desktop PC.**

Isn't *BSD more secure?

A common question, but how true is it really?

This type of question pops up from time to time in security-related newsgroups and mailing lists, often just after someone running Linux has had their PC cracked into and begins to ponder the alternatives. This is one of those types of postings that generates a very lengthy thread. The latest thread on comp.os.linux.security ran to 28 postings (*Why is BSD more secure?*, Jem Berkes, 2002-05-21 21:56:32 PST).

OpenBSD, for instance, used to have the proudly displayed statement "Five years without a remote hole in the default install!" at the top of their home page (www.openbsd.org). Reading between the lines, it seems that "default install" is the important point in this statement because, with this type of installation, very few services are enabled, thereby limiting network access to the machine.

In contrast, most agree that Linux distros are designed for ease of use so the OpenBSD approach of secure by default, which then relies on the user to enable and securely config each service, isn't really appropriate.

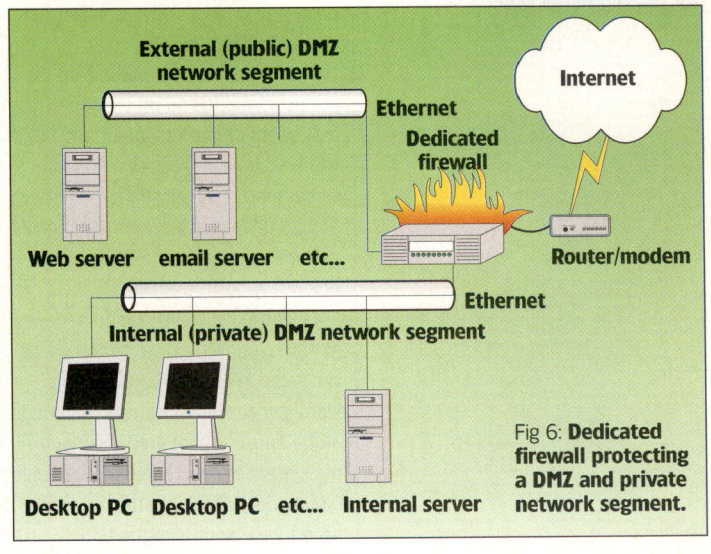


Fig 6: **Dedicated firewall protecting a DMZ and private network segment.**

System maintenance

A firewall isn't much good if it's running buggy software

One of the most common reasons that machines are cracked is because the sysadmin hasn't kept the installed software packages updated.

The security newsgroups and mailing lists are constantly receiving reports of new security vulnerabilities, or holes being discovered in all sorts of software from CGI scripts to OS libraries to fully fledged server programs such as *Apache*.

At the time of writing, over the last couple of weeks there have been security holes found in *OpenSSH*, *BIND* and *Apache* to name but a few. Once these holes are reported, it's a mad race between the systems administrators and the crackers: the crackers are trying to exploit the vulnerability in as many machines as possible before the systems administrators update them, thereby plugging the hole.

I often hear people say things like, "Oh, okay, I'll update it when I get the chance," which is a really dangerous attitude to take. It's dangerous because the cracker community don't take very long to produce a piece of software that hunts down machines running vulnerable versions of a particular program and exploit the documented hole. This is how crackers often crack into machines. Quite often the crackers manage to create and release an exploit on the same day the vulnerability is discovered. These are known as 0day (zero day) exploits.

To find out if a machine is running a vulnerable version of *OpenSSH*, we can simply run *telnet*:

```
# telnet some.server.here.com 22
Trying <IP address here>...
Connected to some.server.here.com.
Escape character is '^]'.
SSH-2.0-OpenSSH_2.9p2
^]
telnet> quit
Connection closed.
```

To make it easier to keep our systems bang up to date, there are tools such as *up2date* on Red Hat and apt-get on Debian that allow a system to be updated automatically every day, or even more often than that. Now there's no excuse.

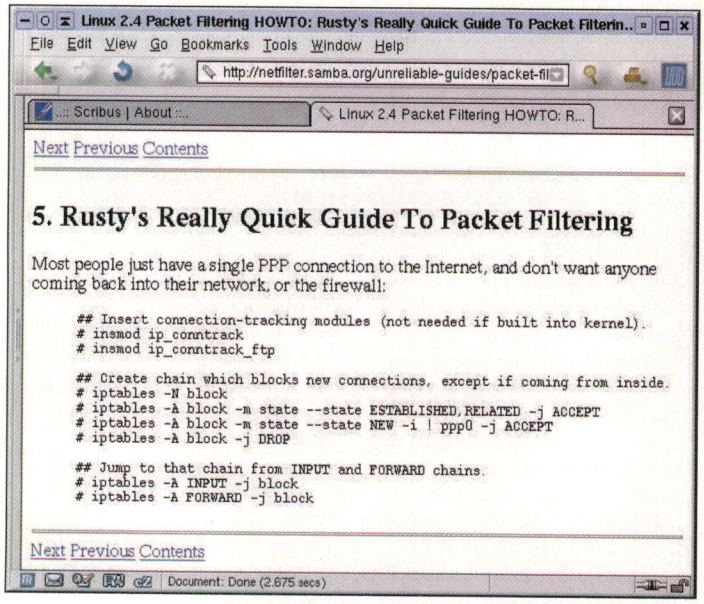

Linux 2.4 Packet Filtering HOWTO: Rusty's Really Quick Guide To Packet Filterin..
File Edit View Go Bookmarks Tools Window Help
http://netfilter.samba.org/unreliable-guides/packet-fil
::: Scribus | About :::
Linux 2.4 Packet Filtering HOWTO: R...
Next Previous Contents
5. Rusty's Really Quick Guide To Packet Filtering
Most people just have a single PPP connection to the Internet, and don't want anyone coming back into their network, or the firewall:
Insert connection-tracking modules (not needed if built into kernel).
insmod ip_conntrack
insmod ip_conntrack_ftp
Create chain which blocks new connections, except if coming from inside.
iptables -N block
iptables -A block -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A block -m state --state NEW -i ! ppp0 -j ACCEPT
iptables -A block -j DROP
Jump to that chain from INPUT and FORWARD chains.
iptables -A INPUT -j block
iptables -A FORWARD -j block
Next Previous Contents
Document: Done (2.675 secs)

Fig 7: **Simple firewall rules for dial-up home user.**

« ■ Are you interested in learning a particular subject such as packet filtering with iptables?

■ How much do you understand the protocols you intend on filtering? This may seem rather less important, but if you wish to write your own firewall script, then you really do need to have a good understanding of the protocols such as ident/auth, or FTP to avoid potentially costly mistakes.

When I first implemented a firewall at home, I only had one Linux box, as illustrated in **Figure 4**. I wanted to learn ipchains (iptables wasn't around at the time) so I got hold of a number of different *ipchains* scripts that people had made available on the web and used them, along with the odd HOWTO, as a guide to compile my own. The main problem with this DIY approach is that unless you really are a network security expert who knows what you're doing, you run the risk of using a less than perfect firewall that may have unplugged security 'holes', or vulnerabilities. For instance, you may have misunderstood how a particular protocol works, thereby leaving a little hole in your defences. This is something that's easily done.

With my bargain-priced second hand P133 I decided to give the GNU GPL-licensed Smoothwall (see the *Resources* box) a try, turning the box into a dedicated firewall. It installed within minutes and was configured during installation with the minimum of required user knowledge. No tweaking of its scripts was necessary, as its default deny policy – where outbound connections were allowed, but incoming connections were denied – served me well.

In fact, it continues to provide rock solid service for my home network with virtually no need for tweaking or administration. Would a custom written script of mine offered me the same level of confidence, in terms of security? Certainly not.

In terms of time, Smoothwall proved to be the most efficient solution for me, although if I had wanted to learn about packet filtering, I wouldn't have got much out of the experience. It all depends on your individual goals and what resources you have available.

Crafting your script

Writing one's own firewall script can take a long time, depending upon the complexity of your requirements, but it does have some advantages. The script is normally very short and simple for a home user, without any publicly available servers/services. It affords you the ability to easily fine tune your script for your own particular needs, which can be considerably more difficult if you weren't the one that wrote it.

It also enables maximum flexibility, which may not be possible with a firewall script creation program or out-of-the-box product. These are the types of requirements you need to consider when choosing the most appropriate method.

To put this into context, **Figure 7** shows just how simple an *iptables* firewall script can be for a dial-up user, which is taken from one of the Netfilter HOWTOs (**http://netfilter.samba.org/unreliable-guides/packet-filtering-HOWTO/packet-filtering-HOWTO.linuxdoc-5.html**).

Whichever route we take in implementing a firewall, we should test it before relaxing! Virtually everyone hates boring testing, but the exercise takes very little time and can be quite interesting. But how can we test our firewall when we're on the inside of it? Well, we can't, but someone else would be able to.

For instance, you could connect to the Internet with your firewall in place and ask one of your friends to run *nmap*, or *xnmap* (a graphical front-end to *nmap*) against it while also connected to the Internet. *nmap* allows you to carry out what's called 'port scanning' against a target host, which is a good method of identifying which ports (services) are open (accepting connections).

If your friend feels uncomfortable about running these tests for you, a simple solution is for you to use *SSH* (*Secure Shell*) to connect to their machine over the Internet and run *xnmap* on their PC to scan your public IP address, with the GUI of *xnmap* appearing and being interacted with from your own screen/keyboard/mouse. There will, of course, be a fairly significant delay in any interaction on a non-broadband connection, but this is

Definitions

Tech terms explained

CROSS-OVER CABLE: Useful where two machines need to be connected with a network connection such as 10BASE-T Ethernet. Instead of buying a hub for connecting the Rx (Receive) and Tx (Transmit) wires from hosts, you can buy or assemble a cross-over cable yourself. Such cables have the Rx and Tx wires crossed over, which simply means that the Tx at one end's connector feeds into the other end's Rx and, similarly, the Rx to the Tx.

BASTION HOST: Another of those terms like firewall that means slightly different things to different people, but generally describes a machine such as a server that provides a limited number of services (like FTP) and is configured with utmost security in mind; locking down as much as possible to reduce the chances of a successful attack.

IPTABLES/IPCHAINS: The user-space programs (as opposed to running inside the kernel-space) that allow users to add permit/reject rules for types of network traffic. *iptables* superceded *ipchains* with the 2.4.x series of kernel.

PORT FORWARDING: One machine – *e.g.* the firewall on the external perimeter – receives traffic for a particular app, *e.g.* HTTP, and forwards it to another box sitting behind it, *e.g.* on the DMZ.

TRANSPARENT PROXY: Certain applications' traffic such as HTTP is redirected through a proxy server such as Squid (www.squid-cache.org) without the necessary knowledge of the user, *e.g.* by a firewall. There is much documentation available on the web, see the mini HOWTO on the LDP website at www.tldp.org/HOWTO/mini/TransparentProxy.html

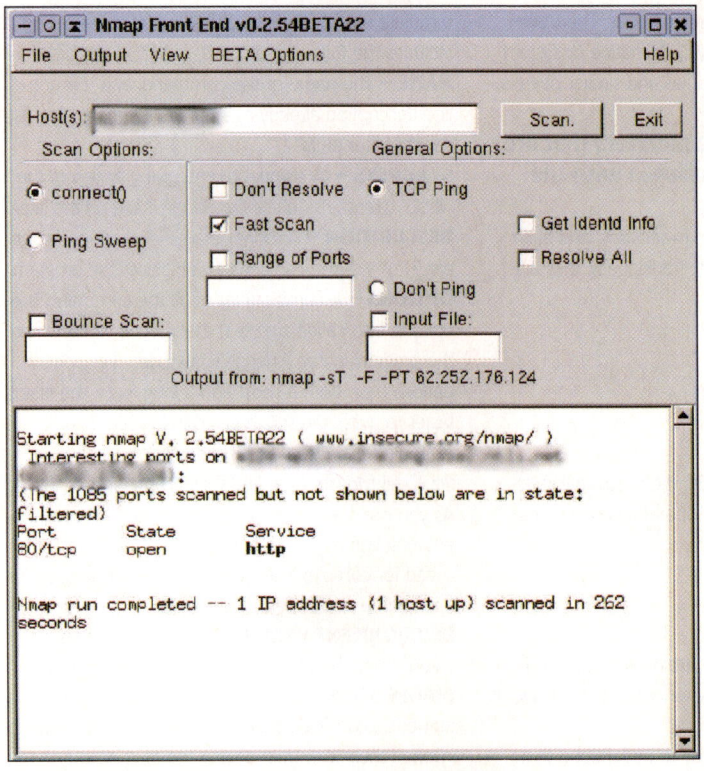


Fig 8: **Using *xnmap* remotely over *SSH* to test the defenses of my firewall.**

actually a very easy way of running remote X clients (*xnmap* in this case) securely and most of the main distributions' default *SSH* configurations enable this X Windows tunnelling through *SSH* by default. However, if they have a firewall in place, you may need to add a rule to the INPUT chain which will allow *SSH* into their machine from yours.

Port scanning is often what a cracker does when seeking out holes in your defenses. Ports marked as 'open', such as FTP, are a prime candidate for attack.

Figure 8 shows a screenshot of *xnmap* being run by me on a remote machine to port scan my dial-up firewall running Smoothwall (only TCP ports in this example).

Bear in mind, though, that results from port scans can show ports being open when, actually, they're not. This can happen when a proxy server sits between the scanner's machine and the target machine.

In the screenshot illustrated in **Figure 8**, my ISP has a **transparent proxy** in place, whereby all HTTP traffic (TCP port 80) is directed through it without an explicit request for this in, say, my web broswer's configuration. The port appears to be open, as if I were running a webserver on my firewall, which I wasn't.

Another important point to take note of is that port scanning is largely frowned upon and may well be disallowed by the terms and conditions imposed by your ISP. If you do wish to carry it out, do make sure you have their permission first, and especially from the person responsible for the target host.

Conclusion

The growth in the Internet has resulted in a similar sized growth in the number of security attacks. Over time, people are realising the extent of the problem and are investing ever increasing amounts of resources to securing their systems. Firewalls are ideal for securing a network or home computer and Linux is an ideal platform to use for this task.

We've only covered network level security, although attacks may be in the form of viruses, or other similarly malicious programs targeted. However, firewalls are generally used as the first line of defence and can even help reduce the effects of some of these other attacks.

For instance, some Microsoft Windows users have recently been targeted with malicious programs that install 'back doors' on the infected machines, which allow crackers to login to the machine and take control of it through this back door (for example an *SSH* server listening on a high-numbered port).

A firewall configured to restrict access from the Internet to the internal private network segment may not be able to prevent the cracker from installing the program in the first place, but at least it prevents them from using the back door to take control of the infected machine. ■

Resources

RECOMMENDED READING

Zwicky, E D, Cooper, S, Chapman, D B: *Building Internet Firewalls* (Second Edition), O'Reilly & Associates.
Bob Toxen: *Real World Linux Security – Intrusion Prevention, Detection and Recovery*, Prentice Hall Inc.
Kirch, O, Dawson, T: *Linux Network Administrator's Guide* (Second Edition), O'Reilly & Associates.

INTERNET RESOURCES

Smoothwall: Linux firewall distribution www.smoothwall.org (GNU GPL) or www.smoothwall.com (non-free licence).

WEBSITES (non exhaustive list)

www.cert.org, www.sans.org, www.dshield.org, www.securityfocus.com, www.incidents.org, www.linuxsecurity.com, www.linux-firewall-tools.com/linux

SECURITY MAILING LISTS

bugtraq@securityfocus.com, focus-linux@securityfocus.com, security & updates lists from your Linux distribution vendor, for example, redhat-watch-list@redhat.com.

HOWTOS, ETC

at www.tldp.org/HOWTO: Firewall-HOWTO.html; Security-HOWTO/; Security-Quickstart-Redhat-HOWTO/, Netfilter/iptables: http://netfilter.samba.org/documentation/

REFERENCES

CERT/CC (2002, 5th April – last update), "CERT/CC Statistics 1988-2002", Available: www.cert.org/stats/.

Internet Software Consortium (ISC) (2002 – copyright date), "Internet Domain Survey", Jan 2002, Available: www.isc.org/ds

Zwicky, E D, Cooper, S, Chapman, D B (2000), *Building Internet Firewalls*, Second Edition, O'Reilly & Associates, California, USA.

What on Earth are... MAN PAGES?

Learning how to learn: Marco Fioretti goes into the most venerable Linux help system to explain how it works, and why it is still a Good Thing™.

Whenever I ask how some program works, the answer is "see the man (page)". Do I have to visit the program author, or leave a message on his pager?

No. You are just being told that before wasting everybody's time you should have read the online *manual* page(s) of that program with the Unix *man* command. Practically, you have to type **man** followed by the command name inside a terminal.

What would I find in a man page?

An explanation of what the command does, and of all the options with which it can be invoked.

From a terminal? In this century? Why not some multi-neural-quantum system? Maybe marketing-oriented and customer-focused?

The man documentation system appeared more than 30 years ago, way before HTML browsers were invented. Many fundamental Unix commands have not changed since then, so they just had their original man page updated. Typing **man** at the prompt may be much faster than starting *Mozilla*.

This is all well and good if I *already* know the command name!

Enter the *apropos* command. It extracts from a database a one-line description of all the commands whose man page contains the word given as argument. The same result can be obtained with **man -k** (**k** is for keyword here).

What if I just want to know what some command is for?

Use the *whatis* command, or just **man -f**: they both return a one-line description of the app's purpose:

```
$ whatis grep
grep (1) – print lines matching a pattern
```

Do man pages exist only for Unix commands then?

No, also for a lot of programming docs. There are man pages for almost every C lib routine or system call, and for many Perl modules and similar objects.

What are those numbers that are sometimes associated with the command name?

The man pages are grouped in sections, and each section has its own directory. Traditionally, they are:

1 User-level commands
2 System calls
3 Library functions
4 Devices and device drivers
5 File formats and conversions
6 Games
7 Various miscellaneous stuff, like macro packages
8 System maintenance and operation commands
9 Kernel Routines
(l) Local
(n) New
(o) Old

Typing **man <number> intro** will output a short description of the kind of material that's dealt with in the corresponding section.

With this in mind, it is easy to understand that **man 1 grep** looks for the grep man page only in section 1, and **grep (1)** means "the grep command, as described in section 1 of the manual". Specifying the number makes the search faster, and sometimes is just needed to avoid ambiguity:

```
whatis socket
socket (2) – Create an endpoint for communication
Socket (3pm) – Load the C socket.h defines and
 structure manipulators
socket (7) – Linux socket interface
socket (n) – Open a TCP network connection
```

Four different meanings of socket, with their own docs.

How is a man page structured?

The actual layout can vary, but the default template hasn't changed much since the beginning. It includes the following sections, usually in this order:

NAME – The name of the command, and the one-line description already mentioned. Example: "**ls** - list directory contents".

SYNOPSYS – "A compendium... giving a view of the whole" (*Webster's College Dictionary*). More on this later.

DESCRIPTION – A more detailed description (what else?) of the purpose and capabilities of the program.

OPTIONS – A complete list of all the command line options, also called *switches*, that can be written after the program name when it is launched. Very often, options have both a short and a long form: the first is faster to *write*, the second is easier to *read*. For example (again from the *ls* man page):

```
-a, --all (do not hide entries starting with .)
```

As you can see, the short form is perfect when entering commands in a hurry, the other is best suited for complex shell scripts which will be used for a long time and/or by many people.

ENVIRONMENT VARIABLES – A list of all the external variables influencing the program behaviour.

DIAGNOSTICS – When programs finish they return a numeric value (**exit status**). The shell or script which invoked them can then decide the next action on the basis of that value. The meanings of the exit states available are explained here.

BUGS, REPORTING BUGS – These two sections report (hopefully) all the known bugs of the corresponding program version, and the email address to which new bug reports should be sent.

FILES – The configuration files read by the program when it starts, and/or the ones where it logs its actions.

SEE ALSO – Links to other commands used by this one, similar commands or other documentation.

Finally, almost all man pages also include a **COPYRIGHT** and/or an **AUTHORS** section.

MANUAL
BUGS
OPTIONS
FILES
NAME
SYNOPSIS
DESCRIPTION

This synopsys looks like a random mix of assorted parentheses, dashes and random letters. Is that supposed to be a human language?

No, not really, although some system administrators speak almost in the same way. The synopsys shows the several ways in which the command can be run, listing all its options and arguments.

Hold on: what is the difference between options and arguments?

Usually the options are the different *things* that a command can *do*: they roughly correspond to the several buttons or menu entries of a GUI program. The arguments are the *objects* (files, users or hardware peripherals) on which the commands *acts*.

If the synopsys is just a list, why didn't they write a plain table?

Because it is essential to know which options can be combined (and in which order) and which can't. For example, the *tar* command can create or read archives, but can't do both things in the same run. A good synopsys contains all these rules, and their syntax is relatively simple. Let's look at the synopsys of *GNU tar*, version *1.13.25* (trimmed for simplicity) :

```
tar c|t|x [-v] [ -f, --file F ][ -Z, --compress]
filename1 [ filename2, ... filenameN ]
directory1 [ directory2, ... directoryN ]
```

The synopsys doesn't says what each parameter is for because that is explained later. Another reason for this structure is that people who already know the command but forgot the name of the same options need read only the synopsys.

The vertical bar (|) indicates mutually exclusive actions: in the example, *tar* can be invoked with the **c** (**Create**) *or* the **t** (**lisT**) *or* the **x** (**eXtract**) option. Square brackets delimit facultative arguments (**-v** generates **verbose** output if present), while commas separate equivalent versions of the same option (**-Z**, **--compress**). The program arguments usually come last: the example above shows that *tar* can act on any combination of one or more files or directories.

This is making me sick! Why do I have to read a PhD thesis when all I really need is a three-line HOWTO?

This is the hardest part to get used to, isn't it? Yes, man pages explain how something *works*, not how it *can be used*: that is left as an exercise for the reader to try out.

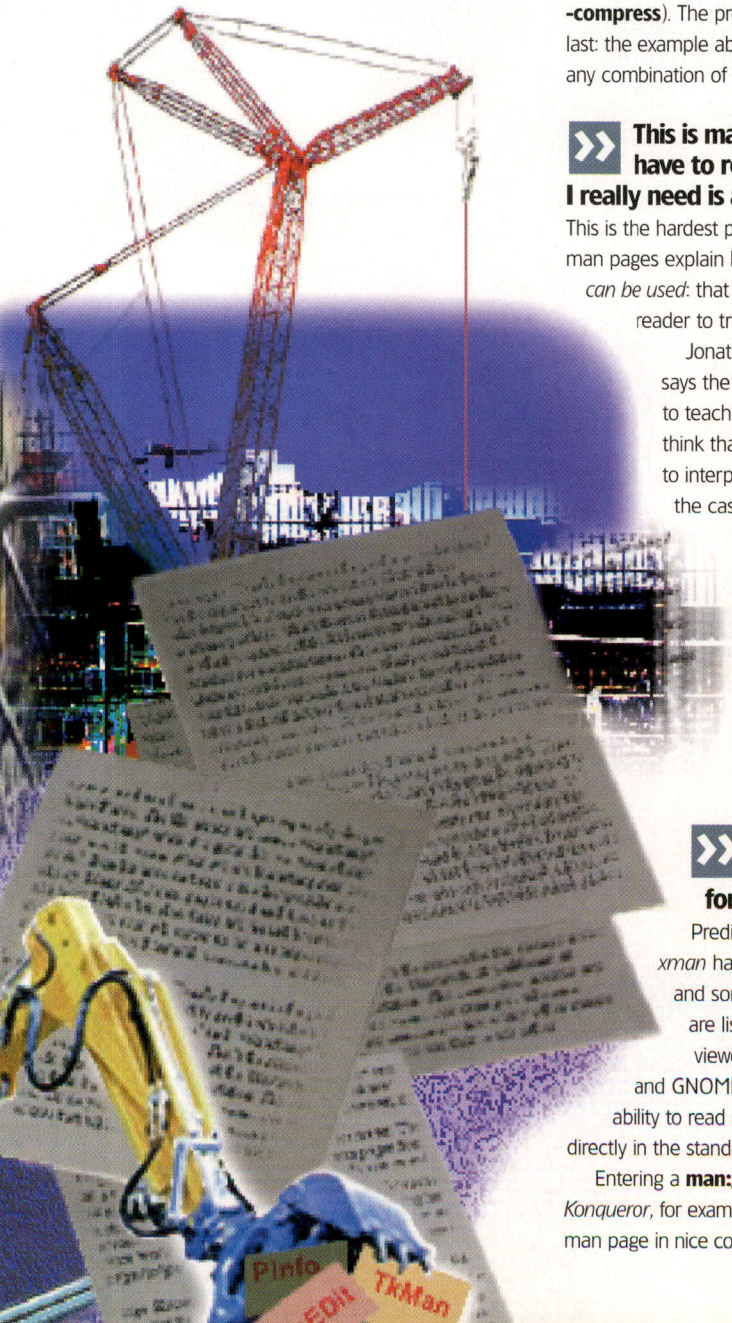
Jonathan Lasser, author of *Think Unix*, says the first Unix thing to learn is, "...how to teach yourself. Most computer people think that everyone is born knowing how to interpret documentation, but it's not the case: it is a learned skill."

This is a Good Thing: command line programs with many options, combined with pipes and scripting, mean that even non-programmers can easily automate almost any task. Ten minutes spent reading a man page can save you many hours in the future.

I like eye candy: isn't there any GUI interface for the man pages?

Predictably, there are several choices. *xman* has been around for a long time and some of the other common options are listed in the box ("man page viewers", above right). Of course, KDE and GNOME have their own viewers, or the ability to read nicely formatted man pages directly in the standard file manager.

Entering a **man://some_command** URL in *Konqueror*, for example, will display the corresponding man page in nice colours.



My browser is up all the time anyway, so can't I just read man pages from it?

Why not? man pages can be converted to plain HTML automatically with *man2html*, included in many distros. The output will preserve all the bold or italic text and generate links to other man pages mentioned in the text. For a fairly complete list of man page converters, including some working in CGI mode, visit **www.w3.org/Tools/Man_faq_filters.html**.

Did somebody already do it? Can I just read them somewhere online?

Yes. You'll find searchable interfaces at **www2.linuxpakistan.net/man.php** and **http://linux.ctyme.com/**, but remember that you may find a version of the command for another Unix.

Emacs brews coffee and irons my shirts: will it also do man pages?

Why yes! In *Emacs*, **Esc X man** (or **manual-entry**) will prompt the man page to load, and display it with all the correct faces. Since **manual-entry** can accept the output of another elisp command, and **current-word** returns the word the point is on, a macro based on "manual-entry (current-word)" will try to display the man page of the selected word.

Emacs is evil. What about vi/Vim?

The pager actually used by man can be specified in the **MANPAGER** environment variable:

```
col -b | view -c 'set ft=man nomod nolist' -
```

will use *vi/Vim* syntax highlighting and nav commands.

Couldn't I just just display the man file with more, less or cat?

To begin with, manual files are compressed to save space. They also include some markup to indent the text, display section headers in bold face and so on. To read a man page you have to decompress it, submit it to the proper formatter (*nroff* or similar) and pipe the output to a pager like *less*. The man command automates this procedure.

Can I save the plain text into a file?

Of course. There are several ways to do it: you could simply pipe the output of **man** to *col* (try **whatis col** to see what it does) and save the output:

```
$ man grep | col -b > grep_man.txt
```

The man command above may even be substituted by *groff* and *zcat* (or *cat* if the source is not compressed) running on the source file:

```
$ zcat grep.1.gz | groff -t -e -mandoc -Tascii
```

How can I print the man pages with bold face and all?

The procedure is similar to the one we just described. All you need to do is to tell *groff* to generate PostScript output instead of ASCII (replacing **-Tascii** with **-Tps**), and pipe the output to the printer:

```
zcat manfile.1 | groff -man -Tps | lpr
```

man page viewers

More sophisticated man interfaces

Pinfo (http://zeus.polsl.gliwice.pl/~pborys/) allows browsing of both *man* and *info* pages with a text interface similar to *lynx*.
TkMan (http://tkman.sourceforge.net) can also browse both kind of pages (warning: some versions force the user to patch *Tk8.0*). *TkMan* can also hyperlink pages among them, search text (plainly or with regular expressions) and do much more.
ManEDit (http://wolfpack.twu.net/ManEdit/) displays man pages, but was also designed specifically to edit them with all the bells and whistles (drag and drop templates, preview and so on).

Man just displays gibberish, or nothing at all!

Sometimes, **man somecommand** might just quit, saying "No manual entry", or format its output incorrectly. In the last case, first check the **LOCALE** settings of your terminal (in Red Hat 8.0, for example, which supports UNICODE by default, you could set the **LANG** variable back to **en_US** or **C** for a correct man page display). Other important environment variables are:

MANPATH – A colon-separated list of the directories actually containing the man pages

MANWIDTH – The width of the man output.

MANPAGER (or, alternatively, **PAGER**) **–** As already explained, if the built-in way used to display the pages doesn't satisfy you, simply change the value of this variable.

LANG – In an ideal world, selecting a certain language during the installation would make *all* the documentation available in that language. In this case, the value of the **LANG** environment variable would tell the man command to first look for the version in that language, showing the English one only if the first is missing

The man system just takes care of itself, right?

Wrong. The correct behaviour of the *whatis* and *man* commands is based on a database created with the **makewhatis(8)** program. This database, and its index, should be rebuilt every time there is a change in the man pages.

Strictly speaking this means any time a program is added, patched, updated or removed, not to mention distribution upgrades. Running **makewhatis** via *cron* is probably the best solution if you update your system quite often.

I have a computer with very little disk space. How do I avoid to install documentation?

Some programs have all their documentation in a separate add-on package, so not installing them solves the problem for good. The main package managers have an option to not install any file marked as documentation (**--excludedocs** for *RPM*).

Of course, if the package author didn't do his or her job properly then that option will accomplish nothing. In such a case, or when installing from a tar archive, you have to remove all the documentation by hand, unless the Makefile has an option to not install any documentation.

I wrote a program: how do I generate a man page for it?

A programmer documenting his code? Allow me to recover from shock! In the meantime, please accept our eternal gratitude, and grab the man page HOWTO, which can be found at:
www.tldp.org/HOWTO/mini/Man-Page.html

What about style?

A poorly written man page will cause almost as many curses and complaints from users as a missing one. Some basic tips to write a good, useful man page are:

- Include some examples, and their *actual* output.
- Test them first, and specify on which Unix/Linux version they worked.
- Remember the difference between users and developers: a man page must explain usage. Of course, do also document the internals of your program, but in a separate document, and point to it from the man page.
- Also, make sure you document *all* the settings in the configuration file (maybe in a separate man page) – see **man man.config**).
- A man page is an official document: be polite, clear, and don't use words or expressions that will be understood only by native English speakers.
- Humour is fine, but in small doses. And run that spell checker, please!
- Check if the source causes any warnings from *groff*, and how the result looks in the graphical viewers mentioned in the "man page viewers" box on this page.
- Format the **NAME** section as explained in the HOWTO, to allow correct indexing.
- Include a valid email address or website for reporting bugs.

You mentioned here and there some so-called "info pages". What are they?

Info pages (or, more exactly, "nodes") are another Unix documentation system. From the end user point of view, the main difference is that they are hyperlinked to each other, so one can browse them very much like web pages.

Of course, trust Unix to complicate things by having at least two unrelated ways to do anything...

The original reason why the *info* system was created is that, while a man page is the condensed instruction manual of some Unix command, an *info* document is a (hypertext) book, which may be just about any possible argument. In practice, the two systems became interlaced and started to point to each other.

Many man pages, especially those for GNU project commands, give a very terse explanation and then just point to the *info* node of the corresponding command for the full documentation.

So, there is a complete Unix online help after all!

Unix has online help all right, but it is sadly (very) far from being complete. As a home assignment, try the following on your favourite distribution:

First, create a list of all your executables (run **find** on all directories in your **$PATH**). Now run **man <command> > /dev/null** on each element of that list count how many times it just says "No manual entry for <command>".

On an almost standard Red Hat 7.3, this little test gave 1191 missing man pages on 2270 binaries. Granted, many programs today have their docs somewhere else in the system, but a man page at least pointing to that somewhere would be nice...

This is great! I can study how a command works on my home Linux box, and show it off tomorrow in our Solaris lab!

Alas, no. The man system (and the knowledge you gain from it) is not completely portable. The first obvious trap is the fact that different versions and/or flavours of Unix carry different versions or implementations of the same command. The *find* command in Solaris 2.7 supports the **-name** option, to find only the files whose names match the given pattern, but the GNU version also offers the **-iname** variant, which is case insensitive. Be warned, especially when moving scripts!

Other pitfalls are in the *man* command itself. Some options are only present on certain platforms (**man -t** doesn't work on HP-UX), or have different names, and the same applies to environment variables. Even some source man pages may not be readable on all platforms.

What are the most useful things I can do about man pages?

First of all, use them to learn. Right after that, always report (both to the program and distribution maintainers) when they are missing or outdated! Even better, write some documentation yourself, and send it to the author. If nothing else, he'll have a template to start from.

There is also a project for this – the Missing Man Pages Project (or m2p2), which you'll find at this site: **www.netmeister.org/misc/m2p2/index.html**. It "aims to provide the man-pages... where the author did not provide an adequate one".

Please contribute to it if at all possible. After all, there's no fun in yelling "read the man pages!" if they are not there... ■

Get more from
Open Source

Contributing to Open Source development doesn't require a degree in programming and getting involved can be a hugely rewarding experience. Richard Drummond outlines how you can become part of a truly radical global movement...

As a user of Linux, you are probably aware that the Linux kernel and all of the tools, libraries and most of the applications that make Linux a usable operating system are Open Source software. But have you ever sat down and considered how all that stuff gets created and who does all the work? Probably not.

Many users of Open Source are quite content merely to be consumers, without ever giving anything back to the Open Source community, and that's a shame because, simply by using this software, you are potentially a valuable contributor.

Active Role

The fact is that Open Source projects depend on people like you and me – as well as high-profile developers such as the creator of Linux, Linus Torvalds, and the pioneering Alan Cox. You don't need to be a masterclass hacker – or indeed be a programmer at all – to take part. Everybody can play an active role, no matter how small. Your contribution might just involve giving feedback to a program's author or submitting a bug report or correcting spelling or grammatical mistakes in the documentation, but whatever way you decide to contribute you are helping to make Open Source software better. And that's what we all want, isn't it?

Before you decide that, yes, this is all great in theory, but you don't know how or you don't have time to get involved, read on...

What's In It For Me?

I thought somebody might ask that, and I know what you are thinking: if people are mugs enough to create all this great software and give it away for free, then more fool them. I'm more than happy to use this free software, but don't expect me to contribute to their folly. If you think like this then, well, you are entitled to your opinion. But you're missing a huge opportunity.

By helping to create free software that can be used and modified by anybody without restrictions, you are helping your fellow man, that it is true. And by becoming an active member of the Open Source community you are taking part in a global movement that is changing the shape of the world economy. These are both noble pursuits, but your reasons for helping Open Source development need not be entirely altruistic. There are real rewards for those that take part.

Fun The driving force behind most Open Source projects is fun: the developers and contributors enjoy doing it. While a lot of projects are started for purely practical reasons, many more grow out of experiments and the play of their founders.

In fact, the Linux kernel itself began this way. Hackers simply love mucking about with computers, finding out how systems work and seeing if they can improve them. If you are similarly inclined, then Open Source can become a stimulating and enjoyable hobby.

Fame Surprising though it may seem, the achievement of fame is one of the significant motivators behind the development of Open Source software. By fame, I don't mean international celebrity status and appearances on TV chat shows, but kudos, recognition and the respect of one's peers.

The Open Source community prizes talent, creativity and hard work. By joining in, you can make a name for yourself, and you could be the one that everybody wants to buy a beer at the next Linux social gathering.

Money I bet that got your attention. Well, actually, very few people earn a living directly from working on Open Source projects and, in general, the amassing of personal wealth is not an important goal for those that do contribute. It is possible, however, to get a paying job working on Open Source, even in these lean times. A select few have been employed on the strength of their efforts in the Open Source arena – Mr Torvalds, Guido van Rossum (the creator of *Python*), and Bruce Perens (Hewlett-Packard Open Source guru) are some famous examples.

"You will be taking part in a global movement that is changing the shape of the world economy."

But even if you can't persuade somebody to give you a salary for working on Open Source, getting involved can improve your job chances, as we shall see in a minute.

Satisfaction I mean here the satisfaction of taking part in the creative process, contributing to a top quality product that people all over the world will use. It is the same sort of satisfaction that a carpenter or blacksmith might experience. After all, software engineering, rather like mathematics, is an art just as much as it is a science.

Friends By becoming an active member of the Open Source community, you are engaging in a social activity. The nature of Open Source development, where

Bruce Perens: now a leading light at Hewlett-Packard thanks to his commitment to Open Source.

»

What is Open Source software?

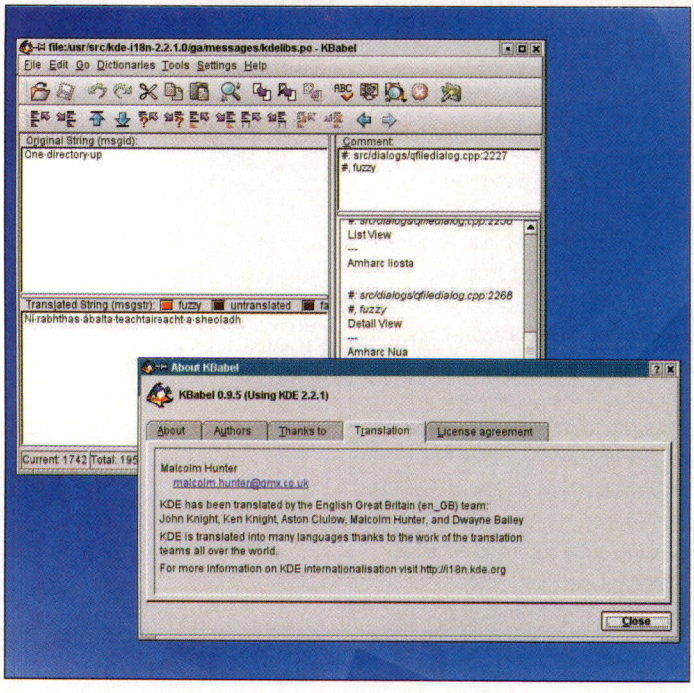
How the release of Linux created a global phenomenon

Open Source software is a term applied to software whose licence gives back the user certain freedoms – the main one is the ability to examine and modify the source code of that software. Depending on the terms of the licence (there are various permutations), you will also be allowed to distribute modified versions of the software.

The father of all Open Source licenses is the GNU GPL or General Public License. A lot of high profile software uses this license, including the Linux kernel. Under the terms of the GPL, you may distribute derived software as long as that software is also licensed under the GPL. See http://www.gnu.org/.

Open Source development in the modern sense began in the 1980s with the birth of the GNU project, but it was not until the release of Linux that the phenomenon really began to take off. This, not surprisingly, coincided with the popularisation of the Internet. Linux is developed using what author and Linux exponent Eric Raymond has since called the Bazaar model because of its seemingly chaotic approach.

The key to successful Bazaar development is to make frequent releases (sometimes even several releases a week) and to involve the users as much as possible (although the distinction between user and developer is often blurred). Users, distributed throughout the world and connected via the Internet, compete to be the first to send in changes, modifications and bug fixes to the project.


file:/usr/src/kde-i18n-2.2.1.0/ga/messages/kdelibs.po - KBabel
File Edit Go Dictionaries Tools Settings Help
Original String (msgid):
One directory up
Comment:
#: src/dialogs/qfiledialog.cpp:2227
#, fuzzy
List View

Amharc liosta
#: src/dialogs/qfiledialog.cpp:2268
#, fuzzy
Detail View

Amharc Nua
Translated String (msgstr): fuzzy untranslated
Ni-rabhthas-abalta-teachtaireacht-a-sheoladh
Current: 1742 Total: 195
About KBabel
KBabel 0.9.5 (Using KDE 2.2.1)
About Authors Thanks to Translation License agreement
Malcolm Hunter
malcolm.hunter@gmx.co.uk
KDE has been translated by the English Great Britain (en_GB) team:
John Knight, Ken Knight, Aston Clulow, Malcolm Hunter, and Dwayne Bailey
KDE is translated into many languages thanks to the work of the translation teams all over the world.
For more information on KDE internationalisation visit http://i18n.kde.org
Close

Translation is a valuable resource and *Kbabel* is a useful tool.

« contributors are geographically distributed and collaborate via the Internet, means that you will come in contact with and communicate with like-minded people from all over the globe. Remember that the Open Source community is famous for its friendliness and its willingness to help others with problems.

Skills Taking part in Open Source development is a terrific opportunity to hone your skills and to learn new ones in whatever role you contribute. If nothing else, it will look good on a CV. Not only that, but potential employers will be impressed with your initiative and enthusiasm because you work on these projects in your spare time. So, even if you can't make money directly from Open Source software, it can help you to gain a better job.

How Do I Join In?

Anybody can join in on Open Source development with very little outlay. The most important thing you need is an interest in a particular project and the will to take part. All the rest you can pick up as you go along.

Other important ingredients you'll need in order to get the most out of contributing are spare time, some skills you wish to share and an appreciation of hacker culture and humour (Eric Raymond's jargon file and other writings can help with the latter – see **http://tuxedo.org/~esr**).

"You'll need to have some spare time, some skills and some appreciation of hacker humour and culture."

How much of the first two you will need depends very much on exactly what role you intend to play. Translating a project's documentation into Urdu, for example, is not a task that everybody is equipped to tackle and is a major undertaking. However, submitting a bug report is something that anybody can do and requires nothing more than a thorough and methodical approach.

Of course, to take part in any role, you will require a computer and an Internet connection, and preferably you should run an Open Source operating system on that machine. Naturally, we prefer Linux, but we're not bigots. Many projects are cross-platform and others exist purely for proprietary OSes such as Windows and MacOS. In short, it is possible to contribute usefully to Open Source development no matter what platform you choose to run.

Last but not least, you will need the legal freedom to contribute. By submitting code or documentation to an Open Source project, you are implicitly agreeing that your work is distributed under whatever licence the project uses. Your own contract of employment may preclude this.

You may not be able to work on any outside IT project or your company may own the copyright for any work that you do even on your own time. If you suspect that a contract of employment may be a problem, check with your legal department at work first – it will save a lot embarrassment and wrangling at a later date for all parties concerned.

How Can I Help?

If you are a programmer, then contributing to Open Source development is relatively easy. Choose a project, download the source code, sit down at your keyboard and start hacking. Simple. (Well, there's actually a bit more to it than that.) But what about the rest of us? What can we do?

The good news is that you don't need to be a propeller-head to get involved with an Open Source project. There are many ways that the less technically-minded can join in, and in fact many roles are often more suited to non-programmers.

Advocacy

The simplest and easiest way we can all contribute to the Open Source cause is advocacy. The success of Open Source software has largely been due to word of mouth, and this is a tradition we can all help to continue.

So, whether you wear a Tux T-shirt when you next go down the pub, or you persuade a friend out of that expensive XP upgrade and get him or her to install Linux instead, or advise

your boss that Linux and Open Source can solve that next IT problem at work more cheaply and efficiently than operating system brand X, a bit of Open Source flag-waving is something we can all do whenever the occasion demands it.

Feedback

An Open Source project cannot exist in a vacuum – by definition it needs users to run, test and contribute for it to progress. The more users a project has, the more successful it is deemed to be. Conversely, project maintainers needs to know people are using their software to affirm that all the effort that goes into the project is worthwhile. Without sufficient interest from users, projects stagnate and many are simply abandoned.

Of course, one could argue that this is how natural selection weeds out the good projects from the bad, but that's exactly the sort of point of view that makes the Open Source community such a stimulating arena for debate. If there is some Open Source software that you use regularly, support the maintainer by giving him or her feedback, even if it's only a virtual pat on the back as a thank you for all the hard work. More constructively, you can make suggestions for improvements.

If there is some aspect of a user interface that you think is clumsily designed, for instance, don't keep it to yourself – let the developers know. If there is some function that you desperately need a program to perform, again, suggest this politely in an email. The developers don't have to act on your suggestions, but you can bet they will read and reply to any criticisms that you make.

Testing

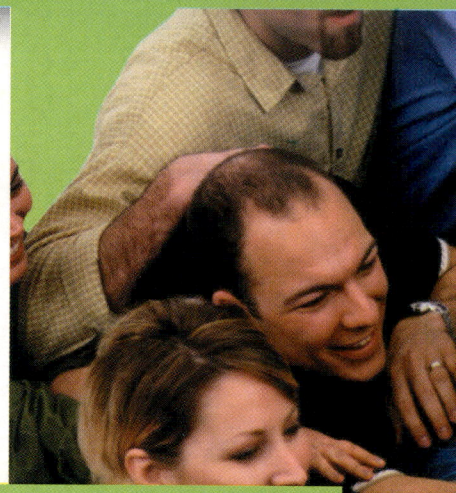
Open source software owes its reputation for robustness to the open way in which bugs are discovered and dealt with. Hence, the reporting of bugs to developers is a crucial ingredient to a project's success.

If you come across a fault in some software that you use, don't just sit back and moan about it – make sure you report it. With Open Source projects, bug reports are taken seriously and are not just swept under the carpet.

The testing of software and the reporting of bugs requires more commitment on behalf of the user than just giving general feedback. To be really useful, bug reports need to give the developer enough information to recreate and diagnose the faults that are reported, so a certain level of technical competence is required to create accurate bug reports. However, this is something we can all learn to do effectively. For more information on how to do it, see the box 'How to submit a bug report' below.

Thankfully, many Open Source projects have automated tools for both the generation and submission of bug reports. Debian Linux has its reportbug tool and an online bugtracking system at **http://bugs.debian.org/**.

Also, the high-profile Open Source web browser, Mozilla, can be compiled to include built-in bug reporting tools and the project also has a »



How to submit a bug report

Correct version numbers and accuracy are the key

A common way of helping an Open Source project is to send in a constructive bug report. The key word here is 'constructive', because, unless the developer is able to make use of the report, it is useless.

A bug report simply describes where and how a program or service does not work as expected. This covers crashes, erroneous or unexpected results and other faults. Sometimes it may be difficult to distinguish bugs from program features, but some common sense and a thorough reading of the documentation will help.

The procedure for submitting a bug report can vary from project to project; some will have automated systems and others will not. In either case, it is important to follow a few careful steps to prepare the perfect bug report.

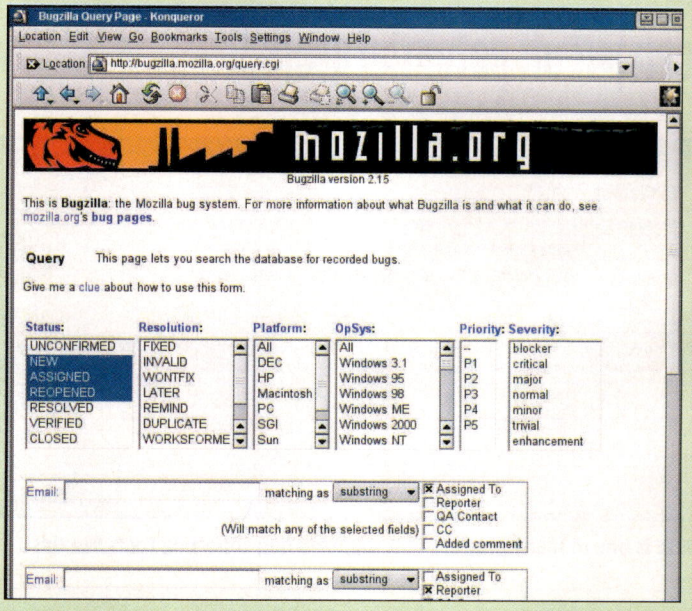
1 Be as clear as possible. If a developer cannot read your report, then they will have no chance of fixing the bug.

2 Include your system details – that is, what distro you are using, the kernel version, compiler version and any other relevant facts. The most important detail to include is the version number of the software that you are submitting the report for. If the software has a back-trace facility, attach any output from that to your report.

3 If possible, state how to reproduce the bug. For example, briefly describe how you were using the software and with what inputs when the bug occurred. If a developer cannot recreate the fault on his system, he will be unable to diagnose the problem.

4 Include any relevant error messages produced by the program. Technical gobbledygook to you may make perfect sense to a developer.

5 Classify the bug according to its severity, to help the developer decide how important it is that the bug should be fixed. Common grades are critical, severe, grave, normal.


Bugzilla Query Page - Konqueror
Location Edit View Go Bookmarks Tools Settings Window Help
Location http://bugzilla.mozilla.org/query.cgi
mozilla.org
Bugzilla version 2.15
This is Bugzilla: the Mozilla bug system. For more information about what Bugzilla is and what it can do, see mozilla.org's bug pages.
Query This page lets you search the database for recorded bugs.
Give me a clue about how to use this form.
Status: UNCONFIRMED NEW ASSIGNED REOPENED RESOLVED VERIFIED CLOSED
Resolution: FIXED INVALID WONTFIX LATER REMIND DUPLICATE WORKSFORME
Platform: All DEC HP Macintosh PC SGI Sun
OpSys: All Windows 3.1 Windows 95 Windows 98 Windows ME Windows 2000 Windows NT
Priority: Severity: -- blocker P1 critical P2 major P3 normal P4 minor P5 trivial enhancement
Email: matching as substring
Assigned To
Reporter
QA Contact
CC
Added comment
(Will match any of the selected fields)
Email: matching as substring
Assigned To
Reporter

The *Bugzilla* online tracking system makes bug reporting easy.

6 Don't submit a wish list or features request as a bug report. Some automated bug-tracking systems can let you assign a severity grade of 'wish list' to a report to request new features. If you do submit such a report, be clear to state that it is a wish list.

A developer's perspective

To get an idea of the developer's side of bug reporting, I spoke to KDE developer and author of *Kdevelop*, Ralf Nolden.

RICHARD DRUMMOND: What factors make a bug report useful to developers?

RALF NOLDEN: Version numbers are the most important thing. The one sending a bug report should say which Linux/UNIX version he uses, the KDE version he uses and which application version he uses. That is very important because sometimes various things can be the cause of the problem or the problem is known to appear on a specific platform.

RD: What things should be avoided in writing a bug report?

RN: The bug report should be as clear as possible about exactly what caused the error and should describe in detail how to reproduce it. The important factor is the reproducibility. The developers only have a chance to fix a bug if it can be reproduced.

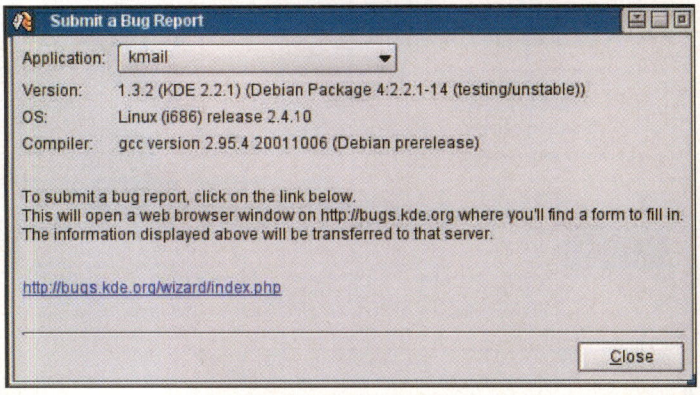
RD: How important are bug reports to you as a developer ?

RN: Bug reports are very important, of course, but they are only that important if they affect a recent version of the program. You see, some things are usually known and are probably fixed. This seems to happen in between releases. The further away the date is from the release date of an application, the more likely it is that the bug has already been discovered – especially if it is a segmentation fault. So, the bug reporter should be aware of that and should look up the bugs.kde.org database in case he is in doubt.

sophisticated online bug tracking system called *Bugzilla* (for more information on this, see the website at **http://bugzilla.mozilla.org/**). The KDE project is another that offers similar bug-reporting facilities.

These types of tools make it easier for users to accurately report bugs and so it is far more likely that they will actually do so. As a result, their importance to the developer cannot be underestimated.

"First impressions count, and a badly designed or poorly maintained website can put off potential users."


Submit a Bug Report
Application: kmail
Version: 1.3.2 (KDE 2.2.1) (Debian Package 4:2.2.1-14 (testing/unstable))
OS: Linux (i686) release 2.4.10
Compiler: gcc version 2.95.4 20011006 (Debian prerelease)
To submit a bug report, click on the link below.
This will open a web browser window on http://bugs.kde.org where you'll find a form to fill in.
The information displayed above will be transferred to that server.
http://bugs.kde.org/wizard/index.php
Close

KDE is one of many projects that supports bug-reporting tools like this.

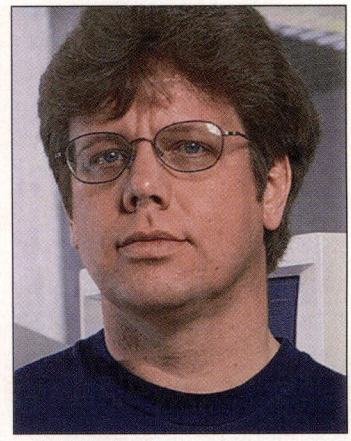
Documentation

Programmers are not always very good at writing documentation for their own software – perhaps because their noses are too close to the grindstone and the problems that average users have may seem trivial to them. So, if there is some software that you use with a poorly written or non-existent manual, why not try to re-write or write it yourself?

Not everybody can write readable and instructive manuals, however. It is skill that needs to be learnt, and you will need a good grasp of both the software in question and the language that you intend to write in. However, don't let that put you off. After all, some documentation is generally better than none at all.

When writing manuals for an Open Source project, you should be careful to adhere to any style or formatting conventions that are in use by that project. Some may prefer docs as plain text, some may want it in HTML and still others may use the DocBook format. (DocBook is an XML/SGML-based format designed specifically for documenting software and is being adopted by many projects – see **http://www.docbook.org**).

You can write documentation with a minimum of tools – any old text editor with a spell-checker will suffice.



Guido van Rossum, creator of the *Python* programming language.

However, for writing HTML or DocBook pages, a more sophisticated editor will help speed things up and catch errors. Most distros ship with an HTML editor, such as *Bluefish* or the excellent *Quanta*, but getting tools to ease DocBook generation can be more problematic. A good XML editor should be able to cope, however. Emacs's PSGML mode works well, as does jEdit's XML plug-in.

If you really want to splash out though, there are some excellent commercial tools available, such as Adobe's Framemaker+SGML.

Translation

Open Source is a truly worldwide phenomenon and, not surprisingly, non-native speakers of English would generally prefer to see their own language displayed on screen rather than English.

Thankfully, most Open Source software supports the GNU *gettext* tools for internationalisation (i18n), and so the task of making software speak to a user in their native language is a job for a translator rather than a programmer. The text strings that a program uses are grouped into a module that is loaded at run time, called a catalogue. The program can then be localised to a particular country and language simply by translating this catalogue.

However, the task of translating a program from one language into another is not as easy as it may seem. You need to be fluent in both languages and understand the idioms that program strings typically use. For example, literally translating 'Save file as' word for word into the target

Getting started with CVS

How to join the global development community

The chaos of Open Source development is often controlled by the source code management tool CVS, the Concurrent Versions System. This lets multiple, distributed developers all work on the same code without stepping on each other's toes. CVS takes care of merging each developer's changes into the main body of code. You may not actually need to use CVS to contribute to your chosen project, but it can be used to keep up with the latest changes to a project, even if you yourself don't submit changes via CVS.

CVS can be a little confusing at first. This mini guide should help you get started. For more information see **http://www.cvshome.org/**. As an example, we will access the KDE project's CVS repository. For other projects the process is similar. The details you will need will generally be available on the project website.

There are several things that need to be done in order to begin using CVS. Typically you will be accessing a source code repository over the Internet using anonymous CVS and the 'pserver' protocol. Anonymous access simply means that you do not need to have an account on the server, and that you will have read-only access.

The first step is to set the CVSROOT shell variable to point to the repository that you want to access. With the standard Bash shell, you can do this with

```
export CVSROOT=:pserver:anonv
mous@anoncvs.kde.org:/home/kde
```

If you only want to access one repository, you can add this line to your .bash_profile so that it is set automatically upon logging on.

Next, we need to log in to the CVS server, like this:

```
cvs login
```

Simply press ENTER when prompted for a password. With anonymous access, you don't need a password. Note that you only need to use login once. In future, the details will be cached in the file .cvspass in your home directory.

To check something out of the repository and store it on the local hard disk, you would use a command like

```
cvs checkout kdelibs
```

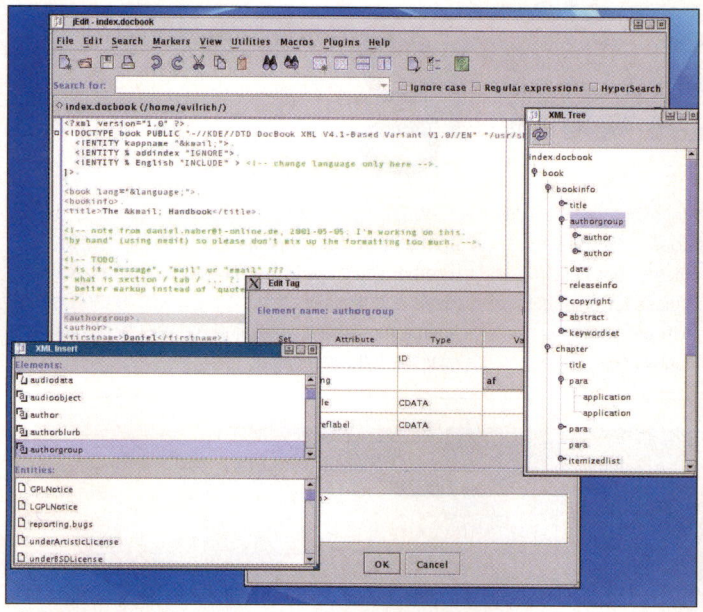
Here 'kdelibs' is the name of the module that you wish to check out.

Many of the CVS commands have aliases, so 'cvs co kdelibs' would achieve the same, and is a little quicker to type.

It is a good idea to periodically update the files on your hard disk to ensure that they are up to date with those in the remote repository. To do this use:

```
cvs update
```

The CVS program itself is reasonably well documented. Use the command **cvs –help-commands** in order to access further documentation.



Using JEdit's XML plug-in helps with DocBook generation.

language will not generally work very well. However, if you feel you are up to the job, then translation is a great way to contribute. Open source projects are calling out for skilled translators, especially for some less well-represented languages.

As with documentation, translation can be performed with a simple text editor, but again tools exist that can do a lot of the spade work for you. For a good example, take a look at the KDE project's *KBabel*.

Illustration

Open Source software, especially GUI-based software, often needs additional graphics. This can include logos, splash screens, icons and the images for toolbar buttons.

The developers themselves may not be the most graphically talented of individuals, so if you are a master (or mistress) of the GIMP, why not lend the developers a hand by offering your artistic talents to a project? Helping to put some visual polish onto an application can make it much more appealing to the user.

Website Design

Because of the way Open Source is developed, a website is typically a point of focus for developers and users. A website may host project news, information, downloads and mailing lists, so it is important to have a site that is clear and easy for the user to navigate.

First impressions count and a badly designed or maintained site can put off potential users from an otherwise worthwhile project. And, often, busy programmers will not have the time or the inclination to devote much effort to the website.

So, if website design is your thing, then taking over the running of the project site can be a useful way in which you can help out, and it's a visible way of stamping your influence on a project.

The level of work involved and the skills required for such a job will vary from project to project. A knowledge of HTML is a minimum, but realistically you will need a server-side language such as PHP too.

Now Do It!

Hopefully we have shown you that everybody has a potential role to play in Open Source development, no matter what their skills, their level of technical ability or the amount of spare time they have to offer.

You don't need to be a highly skilled programmer. Whether you contribute the occasional bug report or re-write a major section of the Linux kernel, all your efforts, however large or small, will be welcomed and highly appreciated.

Now that you know how you can take part and contribute to the Open Source community, it's time for you to actually get out there and do it. You have no excuses... ■

Further reading

We've rounded up some of the best books covering Linux that are currently available, and whether you're new to Linux or a long-time Open Source veteran, we're sure there's something here that you'll find fascinating.

There are biographies which provide surprising insights into the lives of Linux legends Linus Torvalds and Richard Stallman; books covering those who wish to use Linux in a small office or business environment; dedicated help guides for beginners upwards; technical Java and PHP programming guides; and even a history of Linux itself.

Whether you want to learn more about Linux from a technical, programming point of view, or whether you'd like to learn about its origins, ethics, ideals and the key characters involved in the Open Source movement, you'll find the best books around over the following pages.

The Linux Cookbook

In life, as in the command line world, there is always something new to learn, says Biagio Lucini.

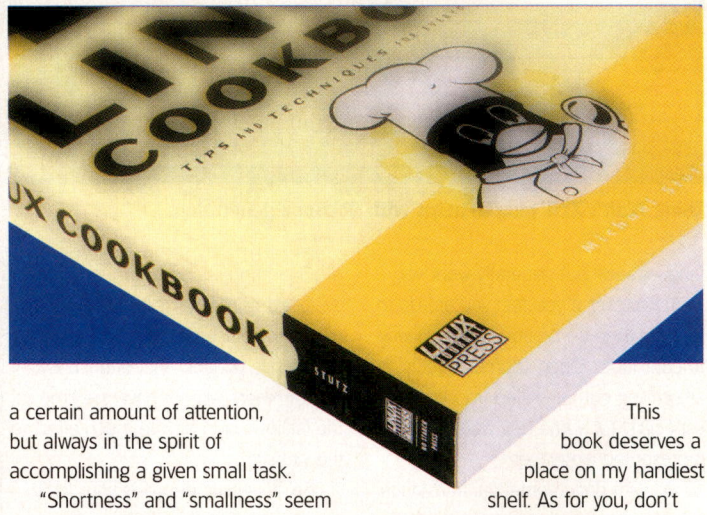
- **AUTHOR** Michael Stutz
- **PUBLISHER** No Starch Press
- **ISBN** 1-886411-48-4
- **PRICE** £23.99

The first time I saw this book with the funny cover featuring a penguin with a chef's hat I thought that it must be another incarnation of the beginner's tutorial on the command line, and after using Unix and derivatives for years, this type of book no longer appeals to me.

However, my first impression could not have been more wrong. *The Linux Cookbook* is indeed focused on the command line, and it is highly recommended to desktop users that are afraid of typing commands in a terminal. However, it uses an approach different from many books I have seen so far. Instead of analytically listing all the possible options of every single command (a procedure that often results in cryptic statements), it just focuses on single everyday tasks, and on what the author reckons are the best ways of achieving them.

The underlying philosophy is the old Unix idea that a given task requires one small but very specialised tool. Here you see this idea at work. A similar approach results in an impressive extension of the ground covered. The simplicity and the efficiency of the proposed solutions are astonishing. As a consequence, even the most experienced shell geek will learn something from this book.

Graphical tools aren't forgotten either, and programs like *The GIMP* get a certain amount of attention, but always in the spirit of accomplishing a given small task.



"Shortness" and "smallness" seem to be the key words of this book: the "recipes" hardly take more than a couple of lines and barely more than five recipes fit one section. As a result, *The Linux Cookbook* has an impressive number of chapters, which range from the fundamental "What Every Linux User Knows" to more advanced topics like text formatting, imaging, sounds and the Internet.

This book deserves a place on my handiest shelf. As for you, don't take just my word for it: take a look yourself at **http://dls.org/cookbook**.

VERDICT

An impressive range of simple but efficient tips and techniques.

RATING 9/10

Linux In The Workplace

Neil Lucock discovers a practical, well written book for Linux beginners.

- **AUTHOR** SSC, Publishers of The Linux Journal
- **PUBLISHER** No Starch Press
- **ISBN** 1-886411-86-7
- **PRICE** $29.95

Although this book's title suggests that it is intended for office use, it's also a very useful book to buy if you are new to Linux. It shows the user how to do various tasks and demonstrates what the different programs are capable of. For example, you get 17 pages on *The GIMP*, and while that's not enough to qualify as a reference book on the subject, it's certainly enough to get you started. I was particularly pleased to see a whole chapter devoted to the *OpenOffice.org* suite. The chapter after it covers *Abiword* and the *KOffice* suite.

There are also details included on using the Internet, email, CD burning tools and, of course, there's also a detailed and comprehensive guide to the user interface.

That's the KDE interface. GNOME is mentioned briefly as an alternative, but they obviously would rather write about KDE. The index does not mention *Nautilus*, *Galeon*, *Evolution* or other key GNOME tools. They all work under KDE so they should be included. If the book has a weakness, this is it.

This book is ideal for the office manager of a company that intends to abandon Windows. She is not a Linux enthusiast, she isn't interested in GPL or computer issues. Someone else configured the computer for her; all she wants to know is how to make it work. To her, it's a tool to do the job, nothing more. Buy her the book and let her get on with it. Once she has learned how to do things, she teaches others and the book is left as a handy reference.

Linux in the Workplace does not delve far into the technical side of Linux, although it does cover shell commands. There are over 300 pages with clear black and white pictures on nearly every page.

The index and program reference (that tells you what each program actually does), combined with intelligent and jargon-free text, make this a good choice for all beginners.

VERDICT

Well written and worth buying. Ought to be included in the boxed Linux distributions as a reference book.

RATING 9/10

Open Source Enterprise Solutions

Jono Bacon dons his pin-striped suit to read a book on Open Source's growing role in enterprise.

- **AUTHOR** Carbone & Stoddard
- **PUBLISHER** Wiley
- **ISBN** 0-471-41744-0
- **PRICE** £35.95

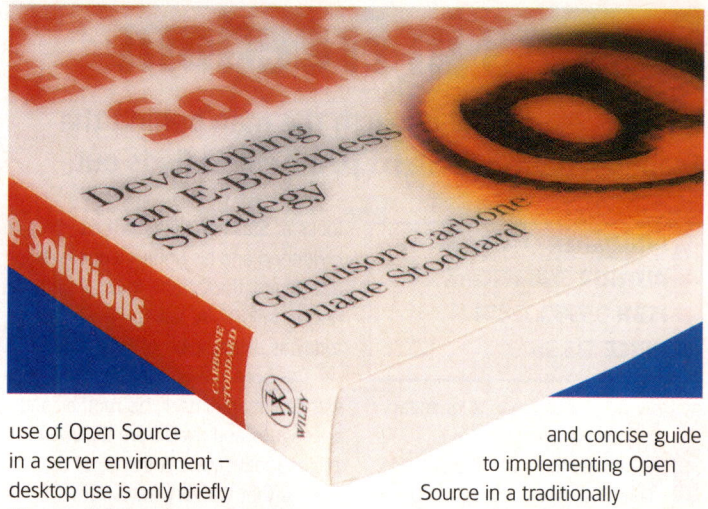
With the recent advances in usage of Linux within corporate environments and enterprises, it is no longer just a hobbyist's operating system. This book aims to show how and where Linux can be used in the business area.

It contains a wide array of topics concerning analysing and implementing Open Source in the enterprise, ranging from details on the various Open Source technologies available to analysis of a strategy, integration of Open Source products and e-commerce.

The sections on integration of a solution, data modelling and protocol implementation are of particular interest. The content is a mix of managerial, technical and analytical information, perhaps occasionally too technical for some IT managers.

The book covers networking, web serving, access to data stores and e-commerce, plus communications. No specific requirement has been placed on the size of the enterprise that an IT solution should fit in, so the book can be used by managers of large IT departments or small offices.

The only real disadvantage is that the book only really focuses on the use of Open Source in a server environment – desktop use is only briefly discussed. Granted, many people feel that desktop use of Linux is not so widespread, but some details on the issues surrounding using Linux on workstations would be useful. More information on interfacing an Open Source solution with a commercial solution like Windows, Solaris, MacOS and so on would also be useful.



Other than these small grumbles, the book is well written, clearly organised and authorative. The authors have done a good job in writing a clear and concise guide to implementing Open Source in a traditionally commercial environment. If you are looking to implement some kind of Open Source in your enterprise, I suggest you give it a look.

VERDICT

Well written and comprehensive analysis of using Open Source in an enterprise. Give it to your IT manager.

RATING 8/10

Just for Fun

Jonathan Wright finds that Linux creator Linus Torvalds is as idiosyncratic in prose as he so often is in person.

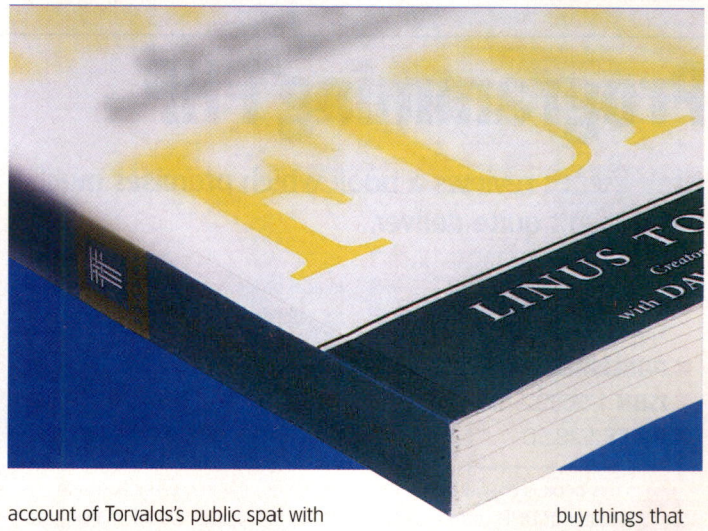
- **PUBLISHER** Texere
- **ISBN** 1-587-99-151-9
- **AUTHORS** Linus Torvalds and David Diamond
- **PRICE** £9.99

One of the more enigmatic characters in the Linux scene is Linus Torvalds himself. He rarely gives interviews and doesn't operate a personality cult like some other Linux luminaries, so the arrival of his biography in 2001 certainly caused a stir of interest.

Co-authored by Red Herring journalist David Diamond, this volume outlines the remarkable Linux story from its head honcho's own unique perspective, and has now been re-issued in paperback format, hence its coverage here. Part autobiography and part sounding board for Linus to bang on about everything from intellectual property and patent laws to the meaning of life (honestly), this is a rare opportunity to get to know one of the world's ultimate geeks, and one of the few people who can legitimately claim to have changed the world.

And so we get the chance to learn that from an early age Linus was fascinated by computers and technology. Hardly surprising, especially when you read Torvalds's portrayal of his parents, Nils and Anna, as left-wing idealists caught up in a heady atmosphere of hippydom.

The early days of Linux are covered in the second part of the book, *Birth of an Operating System*, and while many parts of the story are familiar, it's intriguing to get Torvalds's perspective on events. Nevertheless, it's worth bearing in mind that this is a personal view of what happened and if, for example, you're looking for a balanced account of Torvalds's public spat with Minix creator Andrew Tannenbaum, then this is not the place to get it. (You will find this in *Open Sources: Voices from the Open Source Revolution*, from O'Reilly press.)



The book concludes with Linus sounding off about subjects dear to his heart, and it's here that *Just For Fun* gets, quite frankly, a bit irritating. The problem is that while many of Torvalds's sallies against corporate secrecy are well directed, his faith in free markets and consumer power (suggesting that people simply won't buy things that don't work well) seems to be astonishingly naïve. Perhaps his second book of memoirs, whenever it may be penned, will have a more rounded perspective.

VERDICT

Well it is Linus on Linus, so you probably want to take a look. However, you may be a little disappointed.

RATING 7/10

Free as in Freedom

Can this biography unearth the man behind the Stallman myth? Richard Drummond finds out.

- **PUBLISHER** O'Reilly
- **AUTHOR** Sam Williams
- **ISBN** 0596002874
- **PRICE** £15.95

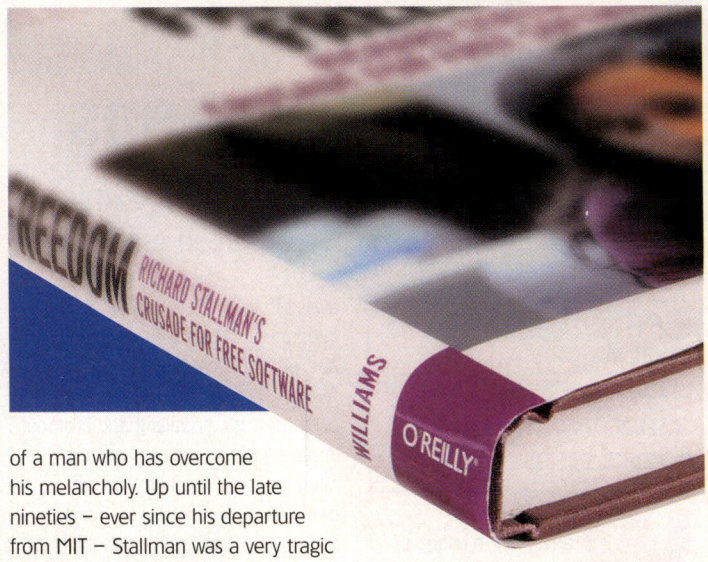
Richard Stallman is legendary in the Open Source community as the creator of *Emacs* and the father of the Free Software movement.

Less positively, he is also famous for his public displays of petulance and his unwillingness to compromise, thus making it all too easy for others to simply dismiss him out of hand. But how much do you really know about Richard Stallman?

Sam Williams's biography takes a sympathetic, but ultimately unpartisan, view of Stallman's life, achievements and failures. It is written for a general audience – and, inevitably, can be simplistic – but Williams isn't afraid to tackle some complex topics. The book looks at Stallman's life from his childhood and his undergraduate days at Harvard, through his idyllic time as a hacker in MIT's AI lab, to his position today as a champion of Free Software.

The book draws from interviews with Stallman himself, his mother, and colleagues and rivals from Stallman's professional career. A theme of the book is that the very traits from Stallman's character that have caused friction in the open source world – his stubbornness and unwillingness to compromise – are the traits that have allowed him to achieve what he has: a person less steadfast in their convictions would have failed.

Not surprisingly, recent events in Stallman's life get the most focus: this is the period that coincides with the explosive growth of the Internet and the rise in popularity of free and Open Source software, and is the best documented. Encouragingly, Williams's account of the 21st century Stallman is



of a man who has overcome his melancholy. Up until the late nineties – ever since his departure from MIT – Stallman was a very tragic figure, the self-styled last true hacker. However, the new Stallman seems to have found his place, educating the world in ethical software: he no longer seems peeved by the Linux usurpers or stuck on the FSF's failure to produce a kernel for his GNU operating system.

Free as in Freedom paints a very human picture of Stallman as a passionate person who loves dancing and food, not just a crusader for ethical software. With the recent fuss in the industry about software licensing, it's easy to forget what Stallman has achieved. This book does well to remind us.

VERDICT

A compelling and human account of the man who started off the free software revolution.

RATING 9/10

Programming PHP

Nick Veitch reviews a book which promises much but doesn't quite deliver.

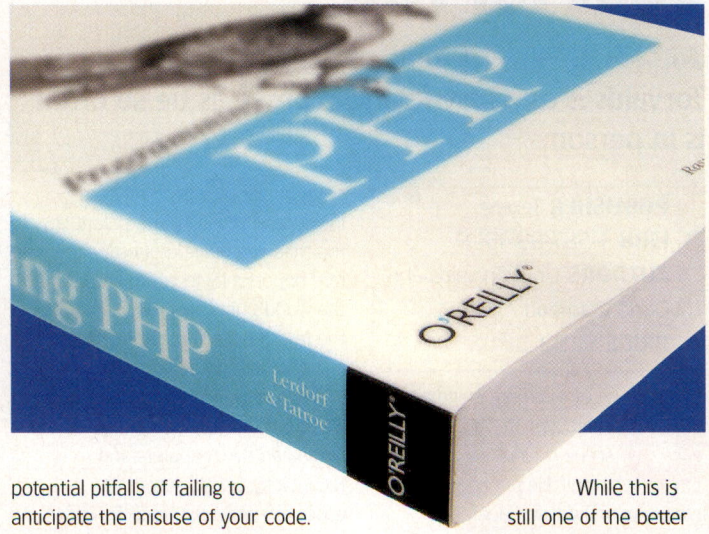
- **AUTHOR** Rasmus Lerdorf and Kevin Tatroe
- **PUBLISHER** O'Reilly
- **ISBN** 1-56592-610-2
- **PRICE** £28.50

As this book is co-written by the creator of PHP, you'd expect to find an authoritative guide, and for the most part you will not be disappointed.

The book starts with the basics of the language – how variables are handled, expressions and operators, functions, arrays and objects. Later chapters cover server side scripting basics, using databases and even the *GD* extension to generate graphics on the fly. There's a section on PDF, and the *lingua franca du jour*, XML.

Within each subject area, concepts are introduced and explained in a logical order, with plenty of examples for reference. However, some of the topics under discussion are dealt with a little too briefly, and you may be left wanting more. The databases chapter is a good example of this. Pretty much everyone using PHP seriously on the web is using it in conjunction with a data source, and I would imagine the PHP/Linux/*MySQL* combo crops up more often than not. Surprisingly then, the databases section lasts for only 13 pages out of 470-odd, and even then only covers the *PEAR DB* interface. The examples are fair enough, but only touch the surface of database interactivity (for example, there's no example shown to check whether the SQL query actually contains any data).

A later chapter deals with securing your PHP code, and although this is only a short section, it includes some wise advice and a few examples of the



potential pitfalls of failing to anticipate the misuse of your code.

Unfortunately, this book seems less than complete. Missing out on the compile-time extensions included as standard (for instance, *MySQL*) means you are going to have to look elsewhere for good info. The function reference for example, only covers the basic PHP functions – not even those that can be specified when building (*MySQL*, for instance). So, it isn't going to replace looking up the PHP homepages for documentation.

While this is still one of the better PHP titles available, it could and should have covered all the subject areas a little more thoroughly.

VERDICT

Probably one of the better PHP books available, although it seems to be a bit rushed in places.

RATING 7/10

Rebel Code

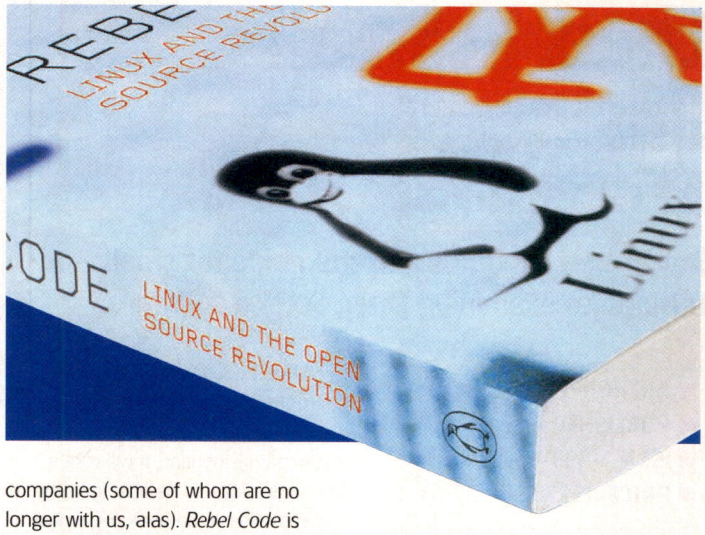
Open Source – the history. Richard Drummond reviews a Penguin guide to Linux's roots.

- **AUTHOR** Glyn Moody
- **PUBLISHER** Penguin Books
- **ISBN** 0140298045
- **PRICE** £7.99

To truly understand the Open Source market today, you need to understand how it got here in the first place. And there's no better guide to the history of the Open Source phenomenon than Glyn Moody's *Rebel Code*, a book which charts the history of the Open Source movement throughout the 1990s. The book has been re-issued in a new paperback edition (rather aptly) by Penguin Books.

Rebel Code follows the Open Source movement from the humble beginning of such projects as the Linux kernel itself, *Apache* and *XFree86* to the dizzying heights reached by Open Source companies like Red Hat and VA Linux – when stocks were soaring and it seemed like the Open Source phenomenon could conquer the world. The first edition of the book was written before last year's Dot Com crash, and so the tone towards the end of the book inevitably had an almost religious zeal. This edition has been brought back down to earth with a new epilogue which gives a more pragmatic overview of today's Open Source market and has more realistic expectations of its future.

The author has based the book on interviews with the key protagonists in the Open Source world, so the reader gets first-hand accounts from people like Linux Torvalds, Larry Wall and Eric Raymond. Accounts of the birth of all the major Open Source projects are featured, as well as the beginnings of some of the major Open Source companies (some of whom are no longer with us, alas). *Rebel Code* is written with the lay person in mind, but is not patronising. Its lively and engaging style clearly reflects the author's passion for the subject matter. However, it doesn't fall into the trap of many popular science accounts: over-dramatising the characters involved.



Rebel Code is an essential read for those new to Open Source, those who haven't actually lived through the history. More than just being an enjoyable and entertaining read, it will help the reader understand the state of the Open Source market today.

VERDICT

An authoritative and entertaining history of the Open Source movement and now even better value for money in this paperback edition.

RATING 9/10

Java in a Nutshell, Fourth Edition

Electronic docs are fine, says Richard Drummond, but a book is hell of a lot more portable.

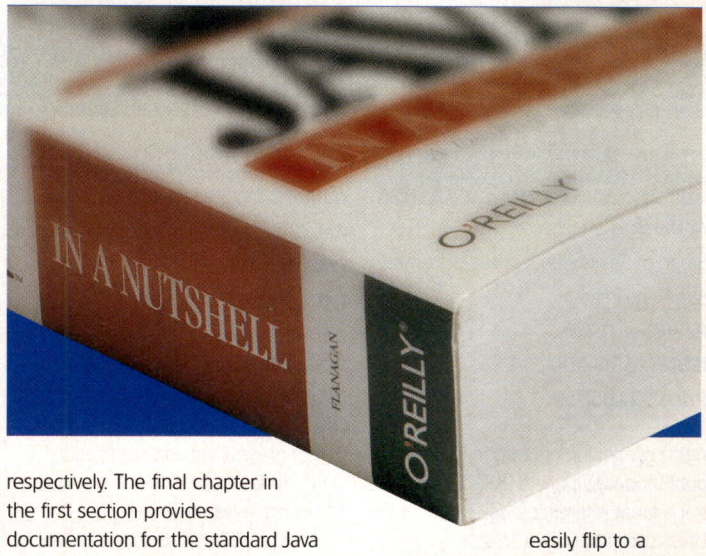
- **AUTHOR** David Flanagan
- **PUBLISHER** O'Reilly
- **ISBN** 0-596-00283-1
- **PRICE** £28.50

Having programming documentation in a searchable, electronic format is valuable, but sometimes there's just no substitute for hard copy. *Java in a Nutshell* fulfils this need, providing a comprehensive, 900-page source of reference to the Java language and the key Java APIs.

The fourth edition has been updated to reflect the changes and new features in version 1.4 of the Java Platform, Standard Edition. This book covers the majority of J2SE 1.4's class library, including the New I/O, XML and Cryptography packages. What's not included are the graphics and GUI packages: *AWT*, *Swing* and *Java 2D* are all notably missing. The first two have previously been covered by a sister volume, *Java Foundation Classes in a Nutshell*, but that one has yet to be updated for J2SE1.4.

The book is divided into two parts. The first introduces the Java language's syntax and features and describes the basic usage of key APIs; the second part forms the main body of the book and contains the reference proper. The first nine chapters provide a guide to Java from first principles. It's not really intended as a tutorial, but it's great when you need to look up how some language feature works.

Chapter four gives a quick but useful overview of the Java class library, while the following two chapters cover security and JavaBeans respectively. The final chapter in the first section provides documentation for the standard Java development tools included with the SDK, such as *javac* and *jarsigner*.



The API reference is machine-generated and typeset directly from the Java APIs themselves, so accuracy should not be a problem. This section is ordered by package name and each class is documented clearly with its fields, constructors and methods all comprehensively covered.

With the fourth edition, however, ease of use has taken a fall: the page-edge marks, which previously let you easily flip to a desired section, have disappeared, and as a result it now takes three times as long to find the class you need.

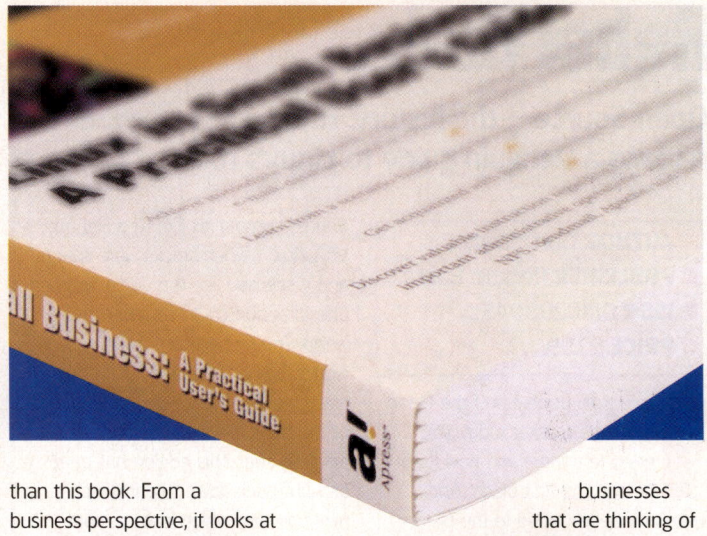
VERDICT

The navigation problems with the fourth edition are disappointing, but this is the only printed reference of its kind covering the Java platform and so is invaluable for Java programmers.

RATING 8/10

Linux in Small Business:
A Practical User's Guide

David Coulson finds a useful guide for small businesses examining Open Source software.



- **AUTHOR** John P. Lathrop
- **PUBLISHER** Apress
- **ISBN** 1-893114-46-1
- **PRICE** US$36.95/£27.00

As Linux is significantly more cost effective than its closed-source counterparts, many businesses are saving money utilising a freely distributable OS, along with taking advantage of the huge number of apps which can be obtained legally from the Internet for no cost. But nothing is completely free. Linux requires a significant amount of understanding to configure, and many services are beyond the capabilities of someone who has never even used Linux, much less installed it before.

While it does start off discussing Linux distros, it is completely based around Red Hat 7.2, so if you're using a different distribution, it will take a little imagination to be able to follow the examples in the book. As almost everything is done using GUIs, it's somewhat difficult to adapt many of the steps for system config, although if you are a Red Hat user, it is very easy to understand. Fortunately, this book manages to do a very good job of hiding the unpleasant text-based config files behind the plethora of GUI admin tools which come with Red Hat.

Everything is covered, from NFS and *Samba* to *Apache* and *sendmail*. However, if you want to to seriously run any of these services, you're going to need something a little more in-depth than this book. From a business perspective, it looks at whether a task can be done in-house or if a consultant should be brought in to set everything up. To see this in a beginners' book is refreshing, although the author may have his own agenda as he runs a consultancy company!

Rather than assuming that a business owner would want to replace everything with Linux, a decent amount is dedicated to integration with systems running Linux, so those using Windows don't have to work out how to use Linux's network capabilities. Even though this book is geared towards businesses that are thinking of switching to Linux, it is a great book for anyone using Red Hat 7.x, and will appeal to those in the home office trying to save money by moving towards Open Source software.

VERDICT

Ideal for a small business trying to save some money, but only if Red Hat is the distribution of choice.

RATING 8/10

SIGNIFICANT OTHERS

LINUX IN A NUTSHELL, 3RD EDITION

Authors: Ellen Siever, Stephen Spainhour, Jessica P. Hekman, Stephen Figgins
Publisher OReilly
ISBN 0-596-00025-1
Price £24.95

Although getting on a bit (it was published way back in 2000), this is still a great reference book for the new Linux user. It covers a little of everything, including administration, desktop use, programming and networking, all in sensibly laid out and easy to navigate sections.

Everything is explained in considerable detail, and although the sections on GNOME and KDE have largely been superseded by advances to these desktop environments over the last few years, there is still a lot of useful information here for the novice Linux user. Only its age prevents a near-perfect score. **VERDICT 7/10**

RUNNING LINUX, 4TH EDITION

Authors: Matt Welsh, Matthias Kalle Dalheimer, Terry Dawson, Lar Kaufman
Publisher OReilly
ISBN 0-596-00272-6
Price £31.95

This book is quite similar to *Linux in a Nutshell* in scope, although it tends to cover more advanced topics and the details of installing and configuring Linux, including information on more advanced networking.

Don't be put off by this if you are a beginner though, because *Running Linux* is well written and most of the topics are explained very clearly.

The book is structured more from the point of view of anticipating problem areas and then explaining how to deal with them. Some of the topics covered include package management, sound hardware and ADLS networking, while there are plenty of server topics for those who are thinking of running *Apache*, *MySQL* or other web services. **VERDICT 9/10**

TUNING AND CUSTOMIZING A LINUX SYSTEM

Author Daniel L Morrill
Publisher Apress
ISBN 1-893115-27-5
Price £24.95

This is a slightly more detailed and specialised book, but it should still be of interest to those who know a little bit about Linux. The goal is to customise and tune a Linux installation for specific tasks, but in order to do this, a lot of the underlying structure of Linux is explored.

Some major distributions are covered in depth, but the majority of the information is applicable to all versions of Linux.

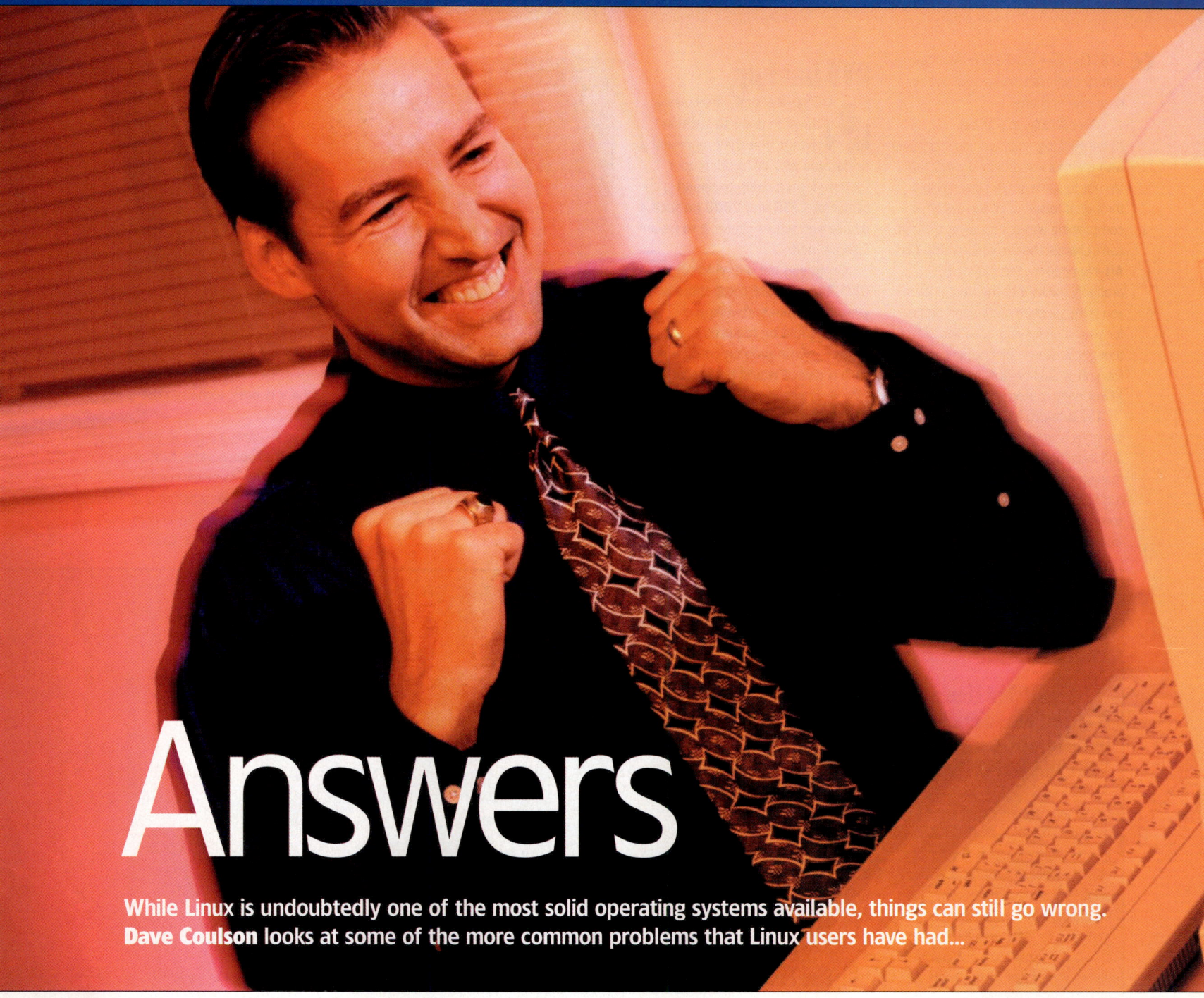
You should get a more detailed idea of what is going on when you install software, and you'll find how to avoid common pitfalls, learn about hardware configuration and more. There is also a complete chapter on building a desktop system which should be very useful. **VERDICT 8/10**

GIMP – THE OFFICIAL HANDBOOK

Authors: Karin Kylander, Olof S. Kylander
Publisher Coriolis
ISBN 1-57610-520-2
Price £34.95

If you've had a play with *The GIMP* and looked at our tutorial you probably have some idea of how powerful a graphics package it is. However, a lot of the time you are only scratching the surface of what you could achieve if only you knew a little more.

With over 900 glossy pages and many colour illustrations, *The Official GIMP Handbook* explains every button tool and menu in depth. Virtually everything is supported with examples and illustrations to help aid understanding, and if you want to get into more advanced usage, there are sections on creating your own scripts, preparing art for print, making animations and more. Overall, this book is extremely well written and is highly recommended. **VERDICT 9/10**



Answers

While Linux is undoubtedly one of the most solid operating systems available, things can still go wrong. Dave Coulson looks at some of the more common problems that Linux users have had...

HARDWARE

External monitor

Q I have Mandrake 8.2 with kernel 2.4.18 and KDE 2.x on my IBM A30p laptop with a Gericom CT1997PF monitor.

I cannot find a way of telling Mandrake to run my monitor at the correct resolution. The laptop has a 1600x1200 TFT screen and X runs fine. However, when I boot with my external monitor attached, I get 640x480 (or possibly 800x600).

I've run *XFSetup*, and Mandrake's own resolution programs, but the screen resolution stays the same. Mandrake's utility tells me to run **KDEINIT: KWIN** after I change the resolution, but then before I can do anything, it reboots the PC! Why and how do I run **KDEINIT**?

I tried inserting a line manually in *XF86Config* but there's been no change. I have even deleted *XF86Config*, and X still runs at the same low res.

It appears that X does not know about my CRT monitor, and does not have a name for it. If I look in *XF86Config*, there is only one monitor section, that for my TFT screen. How do I find out the video display device names in order for me to manually insert a Monitor section into *XF86Config* to enable the high resolutions?

I know the monitor works at these resolutions as I can boot up at 1600x1200 without the monitor attached, and then connect it while X is running. Then the monitor works at 1600x1200!

I am a new user of Linux so I know very little. I tried to fathom out the *insmod*-type utils, but I became lost and still could not find out the name Linux had assigned to my monitor. I have spent months on this on a daily basis.

Do I need to throw out Mandrake and move to another Linux? I hope not as it is otherwise a very easy to use system. Finally, what's the difference between *XF86Config* and *XF86Config-4*?

Owen Widdicombe

A Starting with your last question first, *XF86Config-4* and *XF86Config* are simply alternative names for *XFree86*'s config file. *XFree86* will check a selection of locations for its config – it will first look for *XF86Config-4*, and then for *XF86Config* in each of these. Usually, you would have either one or the other, although many distros symlink »

« *XF86Config-4* to *XF86Config*. If you're getting confused, make sure you only have one of them.

XFree86 logs to /var/log/XFree86.0.log, so when you start X up with the monitor attached, copy this file out to somewhere safe and look through it to see what the system is complaining about. Assuming that you can't change the resolution using **Ctrl-Alt**-Numpad**+/-**, then *XFree86* probably thinks that the frequencies you have supplied are inappropriate for your monitor and used what it thought was best. By checking the log file, you should be able to edit *XF86Config* to stop this happening.

The **kdeinit: kwin** output you are seeing is from KDE when it starts up, and launches the *kwin* window manager. It is not actually telling you to run this – it is actually only supplying some log output so that you know what is going on.

DVD backup

Q DVD-RW units are coming down in price to such an extent that they represent a preferable way of backing up a system. I am looking to spend between £250 and £300 on such a device and would prefer it to be an internal unit.

Could you tell me which of the DVD-RW units are compatible with Linux and what software and drivers would be required?

Any information on configuring such a device would be appreciated.
Mike Bennett

A You're most likely to have success with an internal ATAPI/IDE device, as most external DVD-RW devices are IEEE1394, otherwise known as firewire, which is somewhat questionable under Linux. With an ATAPI device, you can set it up to use the *ide-scsi* module, just like a CD burner, but you need to do things significantly differently in order to be able to write data to the device because DVDs don't handle sessions like CDs do.

```
david@macha:~ (pts/23)
        Option          "CorePointer"
        Option          "Device"                "/dev/misc/psaux"
        Option          "Protocol"              "PS/2"
        Option          "Emulate3Buttons"       "true"
        Option          "ZAxisMapping"          "4 5"
EndSection

Section "InputDevice"
        Identifier      "Generic Mouse"
        Driver          "mouse"
        Option          "SendCoreEvents"        "true"
        Option          "Device"                "/dev/input/mice"
        Option          "Protocol"              "ImPS/2"
        Option          "Emulate3Buttons"       "true"
        Option          "ZAxisMapping"          "4 5"
EndSection

Section "Device"
        Identifier      "Generic Video Card"
        Driver          "i810"
        VideoRam        8192
        Option          "UseFBDev"              "true"
EndSection
                                                66,32-61      58%
```

Adding extra mice to a system running *XFree86 4.x* is easy, although the device must exist when you start X.

You need to apply a patch to your kernel, as well as obtain some tools in order to build a UDF filesystem on the DVD. Once you have a filesystem on the DVD, you can simply mount it and copy files to it. An item of DVD-RW can be

FREQUENTLY ASKED QUESTIONS: STORAGE

FAQ I installed Linux, and it asked me to make lots of different partitions. Can I just install it in one?

Linux can be installed into as few or as many partitions you desire. Usually people have at least two partitions, one being for regular storage, and another for swap space, as swapping to a partition is nicer than swapping to a file on a filesystem.

Having more than one partition allows you to distribute your Linux installation over more than one disk, or allow you to reinstall without destroying the contents of /home. It can also be very useful if you are trying out distros as you could share /home between more than one.

FAQ Which partitions should I create?

The most common directories which have their own partitions are /, /home, /var, /usr and /boot. /home stores user-specific files, so isn't usually destroyed when you reinstall. /usr contains all of the programs and libs you use, and is always distribution-specific. While you could use the / partition for /usr, it's usually more desirable to have a small / partition, and a large /usr partition. /var stores log and spool files, so the filesystem there is generally optimised to store a large number of small files, whereas most others will be storing a small number of much larger files. /boot is only needed if your system has issues booting from beyond 1024 cylinders, and in such a case is created at the beginning of the disk so it doesn't go beyond the 1024 cylinder boundary.

FAQ Can I resize partitions?

The *parted* utility allows partitions to be resized, and you can then resize the filesystem using a utility such as *resize2fs*. However, if you think you will be resizing partitions frequently, or needing to create new ones, using the *Logical Volume Manager* (*LVM*), is a much better choice, as it permits the creation of new Logical Volumes, which are similar to partitions, without having to reboot or restart anything.

FAQ Can I use LVM rather than regular partitions?

Indeed, although your kernel will need to know how to deal with *LVM* when

you first boot up, so keeping / as a regular partition is often a wise choice. Having the rest of the filesystems as LVM logical volumes allows much greater flexibility, and rather than having to limit your partitions to the size of the disk, it will allow you to span more than one disk with a single logical volume, making much better use of the storage space available.

FAQ What about RAID? Can I set up an array with lots of disks?

Linux supports hardware RAID using many controllers, from companies including Intel, Mylex and 3ware. Alternatively, if you want a cheaper option, you can use software RAID and create an array which requires the kernel to handle the synchronisation. While this is slower than using hardware RAID, if you have a lightly loaded system, but require redundancy, then it is a good choice.

For a simple RAID-1 mirror, having two disks mirrored using software RAID can improve read speed and, of course, gives great redundancy should one of the drives bite the dust.

FAQ Does Linux support journaled filesystems?

There is a wide range of journaled filesystems for Linux, including *ext3*, *ReiserFS*, *JFS* and *XFS*. Both *XFS* and

written around 1,000 times, so there are some things you have to do in order to stop it being written by the system when you don't want it to be.

Lots of details, and links to all the packages you need, are at: **http://fy.chalmers.se/~appro/linux/DVD+RW/**

Two mice

I have a Dell Latitude C800 with integrated Glidepoint mouse (pad and nipple), using Mandrake 9.

Can I set up X so that if I plug an external mouse in it will switch to it, and back when I unplug it?

At the moment when I plug the external mouse in the Mandrake hardware thingy pops up and replaces the Glidepoint mouse settings in XF86Config-4 with external mouse settings. I then have to run mousedrake to get the integrated mouse working again.

Can I put two "mouse" entries in XF86Config-4?

From the Linux Format forums

A The pointing devices on most laptops are PS/2 devices, so when you plug a mouse into the PS/2 port on the laptop, the laptop will handle switching from the internal device to the external mouse. There is usually a BIOS option to determine if the internal device will be disabled, or if they will both work together.

If the external mouse is USB, you will need to have the *mousedev* and *input* modules available or compiled into the kernel, then set X up to use /dev/input/misc as an additional mouse in XF86Config:

```
Section "InputDevice"
    Identifier  "Generic Mouse"
    Driver      "mouse"
    Option      "SendCoreEvents"
"true"
    Option      "Device"
"/dev/input/mice"
    Option      "Protocol"   "ImPS/2"
    Option      "Emulate3Buttons"
"true"
    Option      "ZAxisMapping"   "4 5"
EndSection
```

You will need to have both *mousedev* and *input* loaded before XFree86 starts up, otherwise it will die when it tries to open /dev/input/mice. Once it works, you can just plug in a USB mouse and X will automatically allow it to be used in addition to the internal mouse on the laptop.

Video adaptor

I'm trying to play VCD and DVD via *MPlayer* and *Xine*. I have *MPlayer* working, but have to use -vo X11 instead of xv, which I'm told is better.

I did an xvinfo and got the following result:

```
markr@krynn:~/.xine> xvinfo
X-Video Extension version 2.2
screen #0
no adaptors present
```

I'm having difficulty getting any information on why it thinks I have no adaptors. Does anyone have any advice or links to more information where I can read how to get an adaptor configured and installed?

Mark, from the Linux Format Forums

A *xvinfo* is not saying that you have no video adaptors, only that you have no XVideo adaptors. An XVideo adaptor is independent of your video card, although it does require the video card to support it.

You haven't said which video card you have, nor which version of *XFree86* you are running, so you may want to look at upgrading your release of *XFree86* if **www.xfree86.org** suggests that your video card will function as an XVideo adaptor.

If you find that your video card is not supported then there isn't too much you can do, other than swap it out for something else.

Hardware time

Odd one, this. I use *ntpdate* to set my clock when I connect to the Internet. Last weekend, I dialled up to the Internet and *ntpdate* pushed my system clock back an hour to GMT, as expected. However, whenever I reboot my machine, I find the clock has reset to BST (GMT plus one hour). This is really annoying. Can you let me know how I stop this?

From the Linux Format forums

A Each system has two clocks. One of these is a software clock, maintained by the kernel running on the system, while the other is a clock that's maintained by the BIOS, known as the hardware clock. *ntpdate* updates the software clock, so when you reboot the system, it won't be saved.

The utility *hwclock* can be used to write the current software time to the hardware clock by using the **--systohw** switch. Usually, this is performed by distributions when you shut the system down.

Another consideration is the current state of your hardware clock. Generally, the hardware clock should be on UTC, so it won't change when we change our clocks for daylight savings time or back to GMT in the autumn. If your operating system has its time zone set to GMT, but your hardware clock is running on BST, then things will be one hour ahead.

It's worth remembering that if you have your time zones set up correctly on your system, *ntpdate* won't actually change your date when we switch from GMT to BST, or back again. *ntpdate* will always deal with time zones, so you can happily update your GMT system with an NTP server which is running on PST.

If you set your distribution's time zone to GMT or BST, it will not automatically change, but if you set it to 'GB', or if there is an option to select a location, then the system will switch from BST to GMT automatically.

If you don't do this, then you will need to manually change your system's time zone with the *tzconfig* utility when the clocks change, but you will not need to change either the BIOS clock or the software clock as the change of time zone will account for the one hour change.

Graphics drivers

My son bought a G4MX420D-T graphics card, from the Taiwanese company Micro-Star International. The card worked brilliantly in Windows 98 SE but when I tried to boot into Linux it wouldn't work at all, as the GUI logon screen would not appear.

The distro I am using is Mandrake 8.2 and my computer is a Compaq Deskpro, SB series with a Celeron 1.1MHz CPU and 384MB of RAM installed. The hard drive has a »

Linux DVD players, including *Xine*, give much better performance when using the XVideo extension, but not all video cards are supported.

JFS come from large companies, which have ported the filesystem from their own OS, so it is a more tried and tested system, even if not on Linux. *ext3* is simply a journaled version of *ext2*, so if you should need to recover your system and don't have an *ext3* kernel available, you can simply mount it *ext2* and fix it, then remount it *ext3* later.

ReiserFS is a filesystem written for Linux from the ground up, and is popular with people with large arrays, storing hundreds of thousands of files, as it handles lots of small files better than *ext3*.

FAQ Are there any other filesystems which can improve my system?

tmpfs is particularly useful as it is a dynamic version of *ramfs*. Rather than mounting /tmp as a regular filesystem, if we mount it with tmpfs, our system will allocate RAM for it on an as-needed basis. If /tmp doesn't have anything in it, it won't use any memory, so we're not wasting tens of MB of expensive system RAM on nothing.

As *tmpfs* uses RAM for storage, it is significantly quicker than a disk-based filesystem, making it ideal for temporary file storage, although when the system is rebooted, the contents of /tmp will disappear.

capacity of 20GB and it's partitioned with Windows 98 SE in the primary partition.

I have checked the XFree86 site and there isn't any mention of Micro-Star, nor any mention of nVIDIA GeForce4, which is supposed to be the standard of the above-mentioned graphics card.

Micro-Star's packaging claims the card to be compatible with Linux but if this is all rubbish, under New Zealand consumer laws any false claims made by a vendor gives the purchaser the right to return the item and receive a refund.

Have you any suggestions (including sending the graphics card back to the retailer) to get round this problem?

Des Eyre

A The MSI video card is a very popular GeForce4 card so you will be able to use it with nVIDIA's binary driver distribution. You can find this at **www.nvidia.com/view.asp?IO=linux_display_1.0-3123**, although you may want to go in from **www.nvidia.com** to ensure that you get the latest release.

There are RPMs available, along with the kernel module for Mandrake 8.2, so you shouldn't need to compile anything by hand.

Zipping slowly

Q **I have recently obtained the new USB Iomega 750 Zip drive to use on my P733 PC on which I am running SUSE 8.0.**

After plugging everything in, I put a 750MB disk into the device and entered as root:

```
mount -t vfat /dev/sda4 /zip750
```

The device was ready to use.

Unfortunately, the writing speed to the device appears to be very slow – over seven minutes for a 30MB file!

Is there anything that I can do to speed this up? I purchased the device in order to easily back up large files quickly. I appreciate that I am probably not running USB 2.0 for what it is intended and that Linux is not officially supported but other Iomega Zip drives that I have used worked very well under Linux.

This is the first time that I have plugged anything into the USB port and thus I hope that I am not doing anything silly here which is causing the slow write speed.

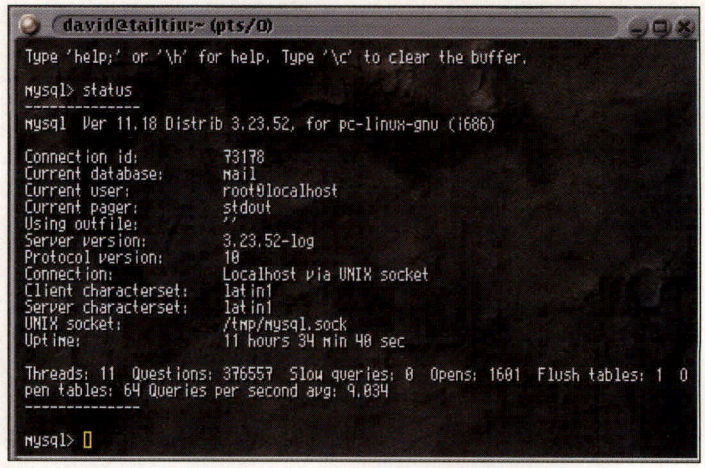
As a comparison, I can write to my internal 250MB Zip at nearly 1MB per second and I can write to my CD-RW nearly as quickly as this when I use the speed=4 option with *cdrecord*:

```
cdrecord -v -eject -force -multi -pad
 -data speed=4 dev=0,0,0
 /tmp/file0000.img
```

Michael Warby

A Regular USB can perform at around 15Mbit/sec, so you should be able to transfer quicker than 73kb/sec. This is likely to be down to the kernel module you are using for USB, so you may wish to switch to *usb-uhci.o*, rather than using *uhci.o*, and see if that performs any better. You may also want to check *dmesg* to ensure that the USB controller is not going crazy with errors, or something else is occurring which could slow things down.

The Linux *OpenGL* library, *Mesa*, requires support from *XFree86* in order to provide the best performance.


david@tailtiu:~ (pts/0)
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> status
mysql Ver 11.18 Distrib 3.23.52, for pc-linux-gnu (i686)
Connection id: 73178
Current database: mail
Current user: root@localhost
Current pager: stdout
Using outfile: ''
Server version: 3.23.52-log
Protocol version: 10
Connection: Localhost via UNIX socket
Client characterset: latin1
Server characterset: latin1
UNIX socket: /tmp/mysql.sock
Uptime: 11 hours 34 min 48 sec
Threads: 11 Questions: 376557 Slow queries: 0 Opens: 1601 Flush tables: 1 O
pen tables: 64 Queries per second avg: 9.034
mysql>

***MySQL* can connect to a server via either TCP or Unix sockets, the latter being more appropriate for local connections.**

If that doesn't help, your next port of call is hardware, so check the cable and ensure that you use the cable that came with the Zip drive, otherwise it may slow itself down in order for data to be transferred reliably.

3D woes

Q **I have used Linux on and off since 1992. I first used Slackware 1.0 beta and then tried all the different distros. I now use a combination of Mandrake 8.2, Slackware 8.1 and Sorcerer. I love Linux for all it is, but I can't seem to shake that WinXP installation. There's still one more frontier as far as I'm concerned, and that is 3D.**

I have a Hercules 3D card based on the nVIDIA TNT2 chipset with a good 32MB of RAM. My Win32 system uses that smoothly and lets me run the newest games in a fair resolution. I recently tried the *Wolfenstein* binaries from idsoftware: full of anticipation I started playing, but to my big disappointment the game was bleaker than a pale hand of ice.

No fog, no lighting, no deep 32-bit textures, no speed, no picturesque resolutions, no FUN!

I run the *3.3.6 XFree86* with the "3D-support". I know my card isn't 100% configured as that XFree86 version is too old. But I run with an identical chipset to the one I support, so it can't be all driver-issues I am facing.

The *3dmesa* libraries are either very incomplete or they have a setup-config-function which I am totally oblivious of. Can you please do a tutorial/piece on 3D graphics at the hardware/driver-level.

I regard myself as a pretty adequate Linux user, but this stuff has left my head twisting. Then again, that may be a direct consequence of the lack of sleep...

Sverre Eldøy

A You may wish to check, using **glxinfo**, that *OpenGL* is enabled correctly on your *XFree86* display. Should *OpenGL* not be working, you may want to just upgrade to *XFree86 4.1.0* and set up the **glx** extension and try again, because *XFree86 3.3.6* is rather old and *Mesa* has improved quite significantly since then.

It does sound quite like *Wolfenstein* decided that it couldn't use *Mesa*, isn't working happily on your system and turned many graphics capabilities off. *Mesa* does not have quite as many features as 3D libraries on Windows, but it is better than it sounds like you are experiencing.

NETWORK

MySQL server

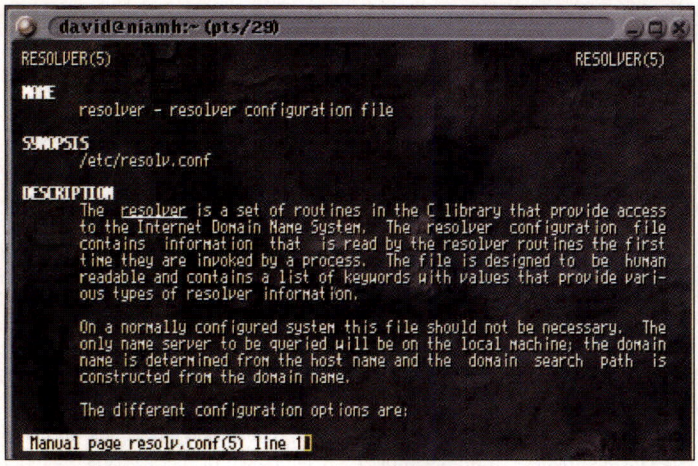
Q **I have installed *MySQL* together with Red Hat 8.0 or Mandrake 9.0. However, when I run *MySQL*, the system prompts this same error code:**

```
Error 2002: Can't connect to local
 MySQL server through socket
 '/var/lib/mysql/mysql.sock' (111).
```

I already did chmod 755 /var/lib/mysql, it would not work! Your help and advice please? Thanks for the attention.

Conrad Fong

A /var/lib/mysql/mysql.sock is a Unix domain socket via which your *MySQL* client will talk with


david@niamh:~ (pts/29)
RESOLVER(5) RESOLVER(5)
NAME
resolver - resolver configuration file
SYNOPSIS
/etc/resolv.conf
DESCRIPTION
The resolver is a set of routines in the C library that provide access to the Internet Domain Name System. The resolver configuration file contains information that is read by the resolver routines the first time they are invoked by a process. The file is designed to be human readable and contains a list of keywords with values that provide various types of resolver information.
On a normally configured system this file should not be necessary. The only name server to be queried will be on the local machine; the domain name is determined from the host name and the domain search path is constructed from the domain name.
The different configuration options are:
Manual page resolv.conf(5) line 1

The *resolver* library uses /etc/resolv.conf to find out which DNS servers it can query.

a local server. In order to be able to use this, you will need to have your *MySQL* server running, and it's likely that this will require you to install another package containing *mysqld*, the server part of *MySQL*.

If you have another *MySQL* server already running on a remote host, you will need to use the **-h** flag in order to set the hostname or IP. More information on the *MySQL* client can be found with **man mysql**.

DNS resolution

Q I need to set up my Linux box so that it can look at another IP on my network as its name server. I can do this in SuSE using *YAST*, although I would rather know which file this is being written to so that I will then be able to edit it myself.

Does anyone know where this is stored? Thanks...

Paul Norris

A The file you are looking for is /etc/resolv.conf, and you'll need to edit the **nameserver** lines within this file. As always, the man pages are useful, so **man 5 resolv.conf** will give you all the information you need.

DHCP

Q I've got SuSE 8 Pro running and have eth0 as an external interface to the net. Eth1 is my internal network serving a couple of PCs.

I installed DHCPD and configured the dhcpd.conf file for my network. However, when I start *dhcpd* I get a message like below:

```
Internet Software Consortium DHCP
Server V3.0.1rc6
Copyright 1995-2001 Internet
Software Consortium.
All rights reserved.
For info, please visit
http://www.isc.org/products/DHCP
Wrote 0 leases to leases file.
No subnet declaration for start
(0.0.0.0).
** Ignoring requests on start. If this is
not what you want, please write a
subnet declaration in your dhcpd.conf
file for the network segment to
which interface start is attached. **
```

I've checked out some docs on the net but can't find anything where a server has two interfaces.

I tried it inside the network with a Linux PC with two NICs and it works fine – it seems to default to eth0 though.

Does anyone know what the command is to make it go to eth1?

From the LXF Forums

A You will need to tell the *dhcpd* daemon to only bind to eth1, and not eth0. *dhcp* can take a list of interfaces as a command line option, so what you need to do is look in /etc/init.d/dhcpd and edit it appropriately. This means that rather than running:

```
dhcpd
```

it runs:

```
dhcpd eth1
```

Many distributions have a **DHCPD_INTERFACE** variable in this file, so you may be able to simply include 'eth1' in there, rather than editing the rest of the *init.d* script.

Your error would also suggest that you have one of your Ethernet devices up, but without an IP address. Without more configuration information, it's impossible to say whether or not this is in error, but if eth1 does not have an IP address in the correct range then *dhcpd* will not function correctly.

File copying

Q Can anyone tell me a way of moving files between networked computers? I have a laptop and desktop hooked up directly to one another and can *ping* and so on. I have been using NFS to mount part of one system's filesystem on the other machine but this is a pain when working with removable media as it won't let me unmount the media without rebooting. It probably would if I knew what it was that needed to be switched off but I don't.

What I am looking for is a simple utility like "send file" in Pocket PCs – just tell one machine to send and the other to receive and wahey! The file is transferred.

Any help is much appreciated but I am new to networking so need something simple!

From the Linux Format forums

A For copying files back and forth between systems, the *scp* utility from *OpenSSH* is often the best choice. This requires that the »

A QUICK REFERENCE TO: DVDs

Watching movies on a computer can be great fun, whether it's having your favourite film on your laptop when taking a trip or using the company's LCD projector to pretend that you are at the cinema watching the latest release. Of course, playing DVDs is not without problems, particularly down to the use of CSS, an encryption method used by studios to stop people copying DVDs digitally.

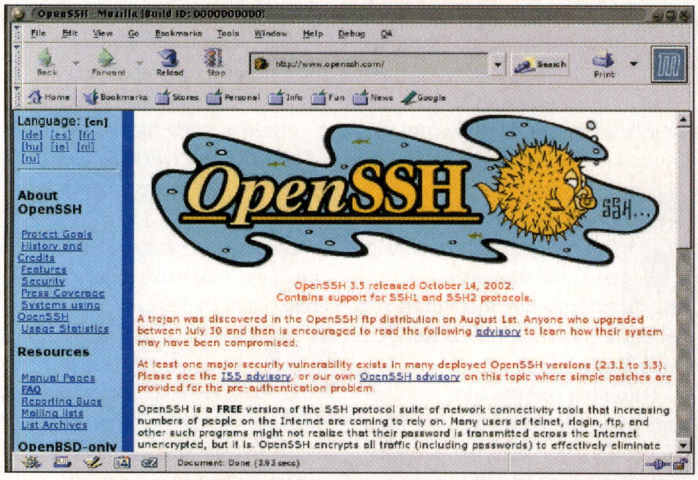
However, this encryption is rather poor, by most standards, and it didn't take a young Norwegian very long to create a crack for it. While this did result in him being prosecuted, we now have an Open Source method of decrypting DVDs on Linux systems, which is usually distributed in the form of *libdecss*.

There are a number of players for Linux which can handle DVDs and DeCSS, the most popular of which are *Xine* (http://xine.sf.net/) and *mplayer* (www.mplayerhq.hu). Both have their pros and cons, but generally, they both work very well and each can take advantage of various DVD capabilities which the other lacks. A particularly problematic DVD feature is interactive menus and *Xine's dvdnav* plug-in (http://dvd.sf.net/) is a good choice, although it does have a number of minor issues when dealing with some DVDs.

Unfortunately, by getting DVD menus to work, we lose out on some speed improvements which the standard d4d player provides. *dvdnav* has been included into the main *Xine* distribution over the last few months, although it does not natively support DeCSS due to legal issues surrounding the distribution of the code within the US. However, individual users are able to install the DeCSS libraries on their own, which *Xine* will make use of.

Many people find *mplayer* to be much faster than *Xine*, not to mention simpler to install, as it doesn't have a whole array of dependencies. Both *mplayer* and *Xine* can play other formats, such as AVIs and DivX files, so we don't need to have half a dozen different apps to play various formats.

It is also possible to copy DVDs and create DivX files, which can be stored locally and played without the DVD being present. A DVD in DivX format may take up around 1GB, although this depends upon compression levels and the resolution used, so we could copy a selection of DVDs and store them all on a large hard disk for quick access. The most popular package for this is *transcode* (www.theorie.physik.uni-goettingen.de/~ostreich/transcode), which can process a large number of video and audio formats into other formats. A DVD can be converted to a format which can be played on almost anything, even a low-end system.


OpenSSH
OpenSSH 3.5 released October 14, 2002.
Contains support for SSH1 and SSH2 protocols.
A trojan was discovered in the OpenSSH ftp distribution on August 1st. Anyone who upgraded between July 30 and then is encouraged to read the following advisory to learn how their system may have been compromised.
At least one major security vulnerability exists in many deployed OpenSSH versions (2.3.1 to 3.3). Please see the ISS advisory, or our own OpenSSH advisory on this topic where simple patches are provided for the pre-authentication problem.
OpenSSH is a FREE version of the SSH protocol suite of network connectivity tools that increasing numbers of people on the Internet are coming to rely on. Many users of telnet, rlogin, ftp, and other such programs might not realize that their password is transmitted across the Internet unencrypted, but it is. OpenSSH encrypts all traffic (including passwords) to effectively eliminate

***scp*, distributed as part of OpenSSH, is a great way to copy files between two systems.**

machine you're copying files to has the *SSH* daemon installed, so that the client can connect to it before copying the files over the secure *SSH* connection.

Each *scp* connection will require authentication, so it will ask for the password on the remote machine in order to authorise the *SSH* connection. This is really annoying when copying a lot of files, so using *SSH* keys is a much better choice, as *SSH* will authenticate without you having to do anything, allowing you to *scp* files back and forth quickly.

You can specify any destination location to write the file to, assuming the user you *SSH* to on the remote system can write to the location.

For large file transfers, *rsync* is also a good choice as it allows copies to be resumed, so if you have a big directory with a load of files, you can stop it and start it whenever you like. *rsync* is generally run over *SSH*, so the need for key-based authentication with *SSH* still exists.

DHCP

Q I have installed Red Hat Linux 7.3 Professional on a K6 500MHz computer with 256MB RAM and a 40GB hard disk. In this computer there is an Ethernet card of this type:

```
D-LINK DFE-530TX PCI Fast Ethernet Adapter (Rev A)
```

This Ethernet card is connected to a Zywall 1 router that give me the IP address and other TCP/IP information. The Zywall 1 router is connected to an ADSL modem.

On the Zywall 1 router there is also a computer connected with Windows 2000 Professional. I use this Windows computer to connect to my company (my office at home). My question is now how I get connected to the Internet from Linux? I have tried to read the enclosed documentation, but I cannot find a useful answer to this question. I hope that you can help me with an answer of this question.

I hope to hear from you.

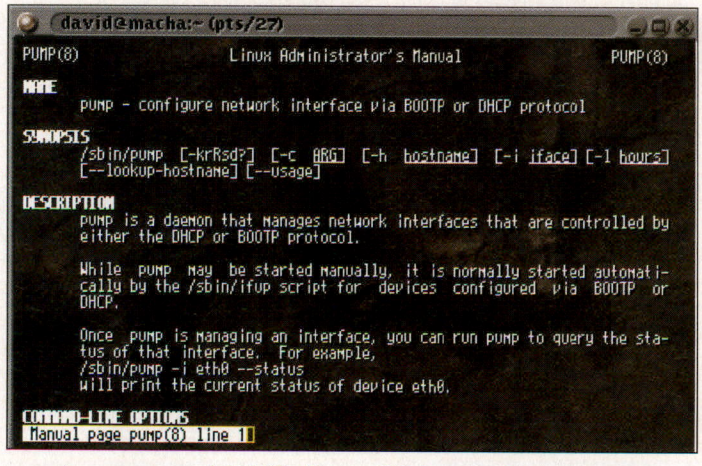
Jan Christensen

A The DFE-530TX NIC uses the *via-rhine.o* kernel module, so you will need to ensure that is either compiled into the kernel or available as a module. Once the module is loaded, you should have an eth0 device which you can configure with *ifconfig*. Depending on how your router is setup, you will need to set the eth0 interface up in Linux with an appropriate IP address and gateway.

You will need to use a DHCP client such as *dhclient* or *pump* to do a DHCP request and get an IP from your router. If you use *pump*, you just need to do:

```
# pump -i eth0
```

This will automatically set everything up, including the default gateway and DNS information. You shouldn't need to do anything else to get Internet access and you should be able to set Red Hat up to automatically perform a DHCP request by using the network configuration tools.


david@macha:~ (pts/27)
PUMP(8) Linux Administrator's Manual PUMP(8)
NAME
pump - configure network interface via BOOTP or DHCP protocol
SYNOPSIS
/sbin/pump [-krRsd?] [-c ARG] [-h hostname] [-i iface] [-l hours] [--lookup-hostname] [--usage]
DESCRIPTION
pump is a daemon that manages network interfaces that are controlled by either the DHCP or BOOTP protocol.
While pump may be started manually, it is normally started automatically by the /sbin/ifup script for devices configured via BOOTP or DHCP.
Once pump is managing an interface, you can run pump to query the status of that interface. For example,
/sbin/pump -i eth0 --status
will print the current status of device eth0.
COMMAND-LINE OPTIONS
Manual page pump(8) line 1

***pump* is a very powerful DHCP client, which is useful if you don't know your network configuration.**

DESKTOP

Swap

Q I'm running SuSE 8.0 on a PIII but despite a listing in fstab and the fact that I specifically made a Swap partition, I don't believe that the Swap memory is being used, which is causing me some problems. Luckily, I have a reasonable amount of RAM (512MB at133MHz). I monitor resource usage and watch as all my RAM is used up but no Swap.

So, how do I make sure that my Swap partition is being used or capable of being used? How do I initialise it without starting from scratch and reinstalling? I've only ever had this problem with SuSE.

Any advice is gratefully received!

Daithidh MacEochaidh

A You can check whether your Swap space is actually being used by running:

```
swapon -s
```

as root. Chances are, your Swap partition is being used by the system, but the kernel doesn't see any point in swapping things to disk.

The vast majority of your memory is probably being filed for file caching and other buffers, rather than memory which has been allocated by processes. The kernel will dump things out of the buffers before resorting to swapping to disk, as this is obviously much more efficient. You can check that Swap is working by running a big application, such as *Mozilla* or *The GIMP*, and leaving the system for a number of days. Generally, memory pages will be swapped out to disk when the application which is running has not accessed those pages for a long period of time, so after a number of days, something which is just sitting there will probably be swapped to disk.

Alternatively, you can attempt to consume a lot of memory by mounting /tmp as a tmpfs file system and using *dd* to create a large file in it, which will use up a lot of your available memory. You could then compile a kernel, or build another large package, as compiling code requires a great deal of memory and the system will likely swap parts of the large file you created out to disk to make room for the memory required by the compiler.

You can always monitor your memory usage with the *free* command and you can quickly check how much Swap space is available, what is used and what parts of your main system memory are being used for buffers and filesystem caching.

```
                total
used        free      shared
buffers     cached
Mem:                 1551592
1490980     60612        0
185860      949224
-/+ buffers/cache:   355896
1195696
Swap:                6291408
1172256     5119152
```

Library versions

Q Firstly I am a complete beginner to Linux. I tried to install the *KDevelop* IDE onto my Mandrake 8.2 system but got the error message:

```
kdevelop <>=S 2.0.2-4mdk
 conflicts with libpng3-1.2.1-6mdk
```

I am then asked to either force or quit, and being a chicken I opt to quit. Can I force the installation without any harm?

Gary Steel

A When there is a conflict, it depends which file they are arguing over. You might want to check that you have the latest *KDevelop* and *libpng3* releases, although you have the Mandrake releases of each package so you would hope that there would be no conflicts with packages built for the same distribution.

You could try to force the install, but make sure that you have a copy of the *libpng* RPM handy just in case things are not happy with whatever *KDevelop* installs. If you don't want to force it, your best bet is to contact whoever built the *KDevelop* package and ask them why you see the conflicts. You can find who built it by using the **rpm -qip <filename>** command.

Print run

Q I have a problem with *OpenOffice.org1.0.1* and printing which I can't seem to get around. I am running SuSE 8.0 on a Toshiba Satellite Pro 4600 laptop, 128MB RAM, 10GB HDD, with *OpenOffice.org1.0.1*.

When I originally installed *OOo* I was running the *lprng* print spooler and could print from *OOo* with no problems. A few weeks ago I decided for various reasons to change my print spooler to *CUPS* using the *gimp-print* and *Foomatic* drivers. *CUPS* installed perfectly and all of the programs including *OOo* printed okay, but *YaST2* warned me that the *CUPS* daemon and the *lprng* daemon would not work together, so I removed *lprng*, just leaving *CUPS*.

Again everything appeared okay until I tried to print from *OOo* and it reported that the *lpr* command did not exist, which was not surprising as I had removed *lprng*.

I cannot figure out how to get *OOo* to print using *CUPS* alone as the *CUPS* documentation gave *lp* and *lpr* as their printing examples. Can you suggest how I can get around this problem?

Many thanks in advance
David M. Swain

A It sounds like you wiped out *lpr* when you removed *lprng*, so the quickest way to solve this issue to simply to reinstall *CUPS*, which should provide *lpr* for you. You can check this works by running **lpr** at the command line, as per the *CUPS* examples.

A QUICK REFERENCE TO: qmail

***Sendmail* is certainly the most well known Open Source mailserver available, but it is by no means the only one, nor the best. Of course, picking a mail server is always a personal choice, but a particularly powerful alternative to *Sendmail* is the hugely popular *qmail*.**

***qmail* has a very simple configuration system which relies on having individual files within /var/qmail/control, rather than one massive configuration file which is a nightmare to maintain.**

***qmail* was designed with scalability and virtual domain hosting in mind, so there is no need for ugly configuration options to make it handle more than one domain, plus users can handle their own mail aliases. This removes some administration control from the postmaster, but it does mean that their job is much easier as all they have to do is point a domain to a particular user, rather than add a whole load of aliases for each domain.**

dot-qmail files are a particularly important part of *qmail*, as these decide how email will be handled by the server once it has been accepted. For example, if we have a domain 'domain.tld' controlled by the user 'bob', bob will get all email for the domain sent through his qmail configuration. If we email 'test@domain.tld', then *qmail* will first check for ~bob/.qmail-test, then for ~bob/.qmail-default.

We can create separate configurations for individual domains, so rather than having 'domain.tld' controlled by 'bob', we can have it controlled by 'bob-domain.tld'. Now *qmail* will check for .~bob/.qmail-domain.tld-test, then for ~bob/.qmail-domain.tld-default and finally for ~bob/.qmail-default. For local mailboxes, *qmail* will use ~bob/.qmail, so if you send a mail to 'bob' on the local machine, *qmail* can handle it differently to a virtual domain.

These dot-qmail files can either forward email to another email account, or insert it into a mailbox. As well as regular mbox format files, *qmail* can also use Maildirs, which are generally considered to be more flexible and less prone to problems than mbox files. If an mbox file becomes corrupt then all your email is gone. If a Maildir entry is corrupt then you lose one email, which is much better.

It is also quite easy to index files in a Maildir, so it's much quicker to pick out an individual message from the Maildir. Not all mail clients support Maildirs, although *mutt* and the popular *Courier IMAP* client will use Maildirs on a *qmail* system.

NETWORK

PCMCIA

Q I am running SuSE 8.0 on a Toshiba S1640CDT laptop, and I am trying to get my PCMCIA Ethernet card running. I'd be grateful for any help you can give.

The card is a Linksys Etherfast 10/100 PCMPC200 v.2. The weird thing is, I bought another card, a sitecom cardbus 10/100, which completely refused to work with the 2.4 kernel, although I tweaked everything in the whole world. After finally giving up on it, I installed the linksys, which worked immediately as eth0. One ifconfig -a later, I had a complete home LAN.

However, after a reinstall (don't ask!) I cannot get the card recognised. No beeps, nothing. I have attached:

```
/etc/pcmcia/config
/etc/pcmcia/config.add_kernel
/etc/pcmcia/network
/etc/pcmcia/network.opts
/etc/pcmcia/config.opts
/etc/pcmcia/config.add_kernel.opts
```

and printouts from *cardctl*:

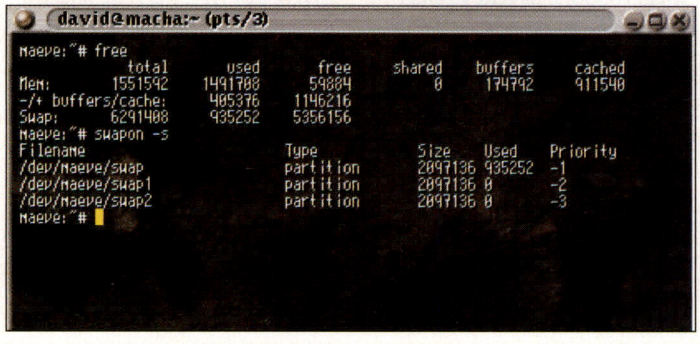
```
toshiba:~/Desktop # cardctl ident
Socket 0:
  no product info available
Socket 1:
  no product info available
  manfid: 0x13d1, 0xff02
toshiba:~/Desktop # cardctl status
Socket 0:
  no card
Socket 1:
  3.3V CardBus card
  function 0: [ready]
Socket 0:
  not configured
Socket 1:
  Vcc 3.3V  Vpp1 3.3V  Vpp2 3.3V
  interface type is "cardbus"
  irq 11 [exclusive] [level]
  function 0:
```

cardmgr gives a null response unless I look at **ps ax**, following a restart of PCMCIA, which gives:

```
 2238 ?   S      0:00
/sbin/cardmgr -k -m
 /lib/modules/2.4.18-4GB/ -n pcmcia
```

ifconfig -a gives the following (I assume sit0 is the firewall. It doesn't seem to be the card, as allocating a network address does not allow connection to the network):

```
toshiba:~/Desktop # ifconfig -a
lo        Link encap:Local Loopback
          inet addr:127.0.0.1
 Mask:255.0.0.0
          inet6 addr: ::1/128
 Scope:Host
          UP LOOPBACK RUNNING
 MTU:16436  Metric:1
          RX packets:94 errors:0
 dropped:0 overruns:0 frame:0
          TX packets:94 errors:0
 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:7048 (6.8 Kb)  TX
 bytes:7048 (6.8 Kb)
sit0      Link encap:IPv6-in-IPv4
          NOARP  MTU:1480  Metric:1
          RX packets:0 errors:0
 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0
 dropped:0 overruns:0 carrier:0
```


david@macha:~ (pts/3)
Maeve:~# free
total used free shared buffers cached
Mem: 1551592 1491708 59884 0 174792 911540
-/+ buffers/cache: 405376 1146216
Swap: 6291408 935252 5356156
Maeve:~# swapon -s
Filename Type Size Used Priority
/dev/Maeve/swap partition 2097136 935252 -1
/dev/Maeve/swap1 partition 2097136 0 -2
/dev/Maeve/swap2 partition 2097136 0 -3
Maeve:~#

There are a number of utilities which can be used to check your Swap space, even if the kernel isn't using it.

FREQUENTLY ASKED QUESTIONS MOZILLA

FAQ What is Mozilla? It's sort of like Netscape, isn't it?

Mozilla was developed using the source code originally used to build *Netscape4*. However, it has practically all been rewritten, so apart from the general look and feel, it's a completely new browser.

Netscape, which is now owned by AOL, has taken the *Mozilla* code and modified it slightly to produce *Netscape 7*. However, *Mozilla* is still in development, so *Mozilla* should be considered over *Netscape 7* when looking for a browser.

FAQ Where can I get Mozilla from?

Mozilla can be downloaded from **www.mozilla.org** and is also packaged by many distros for easy installation. Debian users need only do **apt-get install mozilla** and it installs everything for you. *Mozilla* can be downloaded from mozilla.org in both binary and source releases, and as the *Mozilla* team generate optimised binaries, it's often a good idea to just grab the binary tarballs, as it can take a matter of hours to build *Mozilla* on a reasonably powerful system – it's not something you want to rebuild a few times every day on a PII-500!

FAQ Is it stable, or is it liable to fall over on me and burst into flames?

Mozilla released their *1.0* build quite a while ago and the current stable release is *1.21*. There are also development releases and nightly builds from the code residing in the cvs tree, so you can select the level of 'bleeding edge' you prefer. The current development release is *1.3a*, which may or may not work, depending on what you do with it.

FAQ What can I get done with Mozilla?

Mozilla supports everything *Netscape 4* does, including a web browser, mail client, address book and HTML composer to create websites. There are also a number of other additions, including an IRC client. Due to its development, *Mozilla* also includes various debugging tools for Javascript, HTML and CSS.

FAQ I heard there was lots of cool stuff in Mozilla. What does it do that Netscape doesn't?

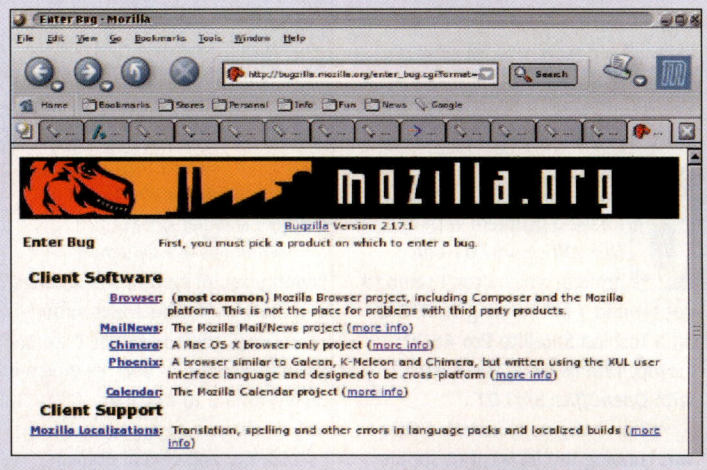
Generally, *Mozilla* is significantly more stable than *Netscape 4*, and the rendering engine known as *Gecko* is completely compliant with the HTML, XHTML and CSS standards, as well as being much faster than the HTML rendering system which was used with *Netscape 4*.

There are a number of HTML and CSS tests which are available with *Mozilla* to show off its swanky rendering capabilities, and all the fancy stuff it can do with CSS to make your sites even more exciting.

FAQ Are there any third-party additions available for Mozilla?

There is a multitude of third-party projects for *Mozilla*, many of which can be found over at **www.mozdev.org**. Many of these add extra capabilities to *Mozilla*, or extend upon those which are currently available. Indeed, a number of mozdev.org projects have been merged into the main *Mozilla* tree, exposing these projects to a much wider audience.

Hop on over to mozdev.org and take a look at the top 50 projects and see if there are any which interest you. Some are quite silly and don't do anything particularly useful, but others


Enter Bug - Mozilla
mozilla.org
Bugzilla Version 2.17.1
Enter Bug
First, you must pick a product on which to enter a bug.
Client Software
Browser: (most common) Mozilla Browser project, including Composer and the Mozilla platform. This is not the place for problems with third party products.
MailNews: The Mozilla Mail/News project (more info)
Chimera: A Mac OS X browser-only project (more info)
Phoenix: A browser similar to Galeon, K-Meleon and Chimera, but written using the XUL user interface language and designed to be cross-platform (more info)
Calendar: The Mozilla Calendar project (more info)
Client Support
Mozilla localizations: Translation, spelling and other errors in language packs and localized builds (more info)

***Mozilla* (www.mozilla.org) is a very powerful web browser, which supports all of the current standards for HTML and CSS, making it a great choice for developers**

collisions:0 txqueuelen:0
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

Please help, as having spent hundreds of hours trying to learn Linux, and tons of money buying the distros, books and compatible hardware, I don't want to be forced to drop Linux at this stage.

I know the thing works, but I can't reproduce how it did so.
Paul Hawkes

A When you insert a PCMCIA card, the system should make a couple of beeps. If the last beep is high pitched then the card will generally have installed correctly. If it is low pitched, then there is a problem. If you get no beeps at all, then it would suggest that something isn't very happy with your PCMCIA set up.

Many 2.4 users find that they have more success using the *pcmcia-cs* distribution of modules, rather than the kernel PCMCIA system. You did not indicate which you were using, either now or before you reinstalled, so if you are currently using the kernel modules then you might want to remove them and build the *pcmcia-cs* package from source against your current kernel.

By looking at /var/log/messages, or /var/log/syslog, you can spot exactly what *cardmgr* is doing and which module it is trying to load when you insert the card.

The sit0 interface is used to create IPv6 tunnels over existing IPv4 networks, so rest assured that the problem isn't anything to do with your PCMCIA config or its networking.

GENERAL

Shared memory

Q I am using RH 8.0. I was using the command *df* to see my free space and that's when I notice a directory called /dev/shm. I checked fstab and this is the information I got:

none /dev/shm tmpfs defaults 0 0

I want to know what this is. Is it my Swap partition?
From the LXF forums.

A SHM is for Shared Memory, which is memory allocations shared between processes on the system. tmpfs is a filesystem which grows when required and is allocated out of system memory, rather than living on a physical disk.

A process can create a file in /dev/shm and another can read it, assuming the file access properties are appropriate for such an operation, allowing data to be shared between devices. This can be performed with a regular filesystem, but it will be much slower as it will need to be written to disk, then read back from disk for the other process.

tmpfs is also often used for /tmp and /var/tmp, as it's much quicker for temporary file storage than a /tmp mounted from a disk partition. tmpfs also used to be known as shmfs, because it was originally developed for use with /dev/shm.

You can find more information on tmpfs by checking Documentation/filesystems/tmpfs.txt within the Linux kernel source tree.

are extras which make you wonder how you did without them before.

FAQ Oops! Mozilla has died on me. How do I report this?

The *Mozilla* guys developed a bug-tracking system known as *Bugzilla* which holds all of the *Mozilla* bugs. Anyone can submit bugs to *Bugzilla* at **http://bugzilla.mozilla.org/enter_bug.cgi?format=guided** or search for existing bugs and find out the status of them with a particular Mozilla build.

In order to submit problems to *Bugzilla*, you will need to register with the bugzilla.mozilla.org site, and then you can post bugs to the system. As always, it's a good idea to include as much detail as you can in order to ensure that your bug can be reproduced and studied by those who can fix it.

FAQ Is Mozilla comparable to Evolution in any way?

Not at all. *Evolution* is a PIM, whereas *Mozilla* is a web browser. *Evolution* is more comparable to something like Microsoft *Outlook*, rather than *Mozilla*.

Of course, it is perfectly possibly for you to use both *Mozilla* and *Evolution*, effectively replacing the need to have *Internet Explorer* and *Outlook* available.

Linux for Macs

Q I have an old Mac PowerPC 9500/120 for which I could try out Linux. However, I am not very sure about Linux for the Mac, even though I have bought the latest *Linux Format* mag.

I'd like to ask a few questions to help me on my way:

1 Is there a version of Linux suitable for this Mac? If so, how do I get it?

2 I've read somewhere about having to partition HDs to run Linux. Would I have to do this on the Mac?

3 I'm not a programmer so if the answer to the first question is yes, does it come with simple installation instructions? Or can I get these elsewhere on the Internet?

4 On the DVD for the December 2002 issue of *Linux Format* magazine there is *OpenOffice.org*. If I do manage to install Linux on my Mac, will I be able to install this software as well?

I'm sorry if all this is a bit basic but I am keen to try out Linux on both my Mac and PC!

Keith A Rogers

A Many Linux distributions are available for the PowerPC (PPC) architecture, including Red Hat, Debian and SuSE. If there is a particular distribution you prefer to use, then see if there is a PPC build of it that you can use. You do need to partition your hard disk so you can continue to use MacOS, but this is all part of the installer.

As for *OpenOffice.org*, you will need to either obtain PPC binaries or build it by hand using the source code. You can easily check whether a binary release is for PPC as it will contain 'ppc' in the filename. If it's for an Intel/AMD system, it will be called 'i386' or 'i686', or indeed any 'ix86' architecture available.

For more on why it's worth installing *OOo*, check out the tutorials in this magazine, starting on page 16.

Fragmentation

Q It's been three month since I started to use Linux. In all the books and magazines I read, I cannot find any hint whether I should do a regular *scandisk* or defrag on my Mandrake box.

What are the regular tasks I should do to keep my system happy? How do I go about performing them? And if no such tasks are actually needed, why not?

Bruno

A *Ext2* is generally smarter than *FAT32*, so there is no need to defragment the filesystem frequently. It will automatically rearrange things to improve throughput and to avoid huge gaps in the filesystem. The Linux version of *scandisk* is *fsck*, which is generally run once every ten or so reboots, meaning you don't have to run it manually and you can usually just forget about it.

Alternatively, you can run a journalled filesystem, such as *ext3*, *XFS* or *ReiserFS*, which is checked quickly at every mount so that anything which wasn't written correctly when the system was rebooted can be recovered, or at least it will try to.

Every day Linux runs a number of scripts which are located in /etc/cron.daily/, including logfile rotation scripts and a system which uses *updatedb* to index every file on the system so that *locate* will work correctly. There is often little need to change these scripts, although if the system is frequently off during the night it may be a good idea to change /etc/crontab to run the scripts at a different time, as many distributions run their daily *cron* jobs somewhere between 2am and 6am.

More info on modifying your crontab can be found with **man 5 crontab**.

Boot choice

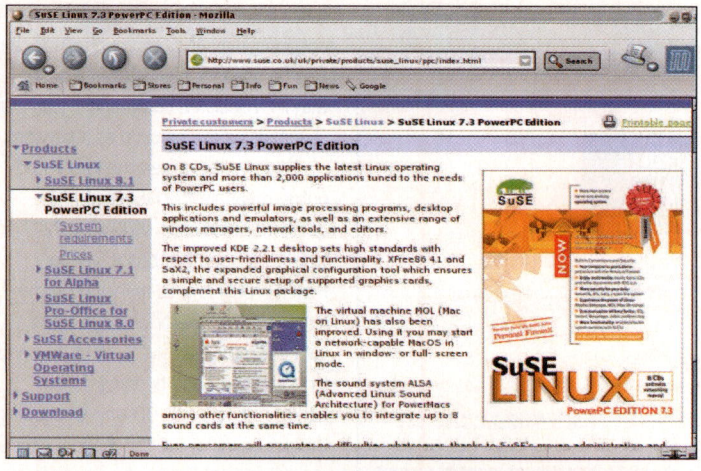
Q I dual-boot Linux and Windows on a server that has no monitor. I use *Lilo's* LOCK option to make the system boot whatever was run last. If I want to *change* OS, I use an attached Wyse serial terminal to intercept *Lilo* before it boots the default (*i.e.* the LOCKed) OS.

What I'd really like, is to be able to nominate the OS to boot before rebooting. Of course, such a facility would have to work under Linux, Windows and whatever other OSs I might want to run.

Have you come across such an application? Would there be any benefit, with regard to the above requirement, in adopting *GRUB* instead of *Lilo*?

A The only option we can think of is to modify the **default=** line in lilo.conf, then rewrite the *Lilo* block in the MBR. There is no nice way to do this via any OS other than Linux, so you're a little out of luck there. Of course, as you have the console available on a serial port, you could quite easily create a script to wait for the *Lilo* prompt and then feed it a new command line.

Quite how you implement this would depend upon how you choose to select the OSs, although the *expect* utility could come in quite handy.



SuSE Linux is one of many Linux distributions which provides a PowerPC build of their distribution.

Timed out

Q I am trying to schedule a daily backup with crontab in Mandrake 8.2. I have edited my crontab -e. The job starts and then stops 30 seconds later. I have tried writing a script and pointing *cron* at it but the same happens.

I have checked Google for too long and have seen nothing relevant. The entry in crontab is:

```
* 18 * * * tar —gzip -cvf
/home/phil/backup/daily_working.tar
/home/phil/working
```

It is driving me to distraction – please can someone help! It seems so simple but I must be doing something wrong.

From the Linux Format forums

A Your cronjob will run every minute when the hour is 6pm, so rather than running it once, you will actually be running it 60 times. You can modify your crontab just to run it once at 6pm with:

```
0 18 * * * tar cvfz /home/phil/backup/
daily_working.tar.gz
/home/phil/working
```

When the job has finished, it will email you the output to your system mailbox, usually in **$MAIL**, so you can check the output from that. You will want to check that it creates /home/phil/backup/daily_working.tar.gz, and also check upon the contents of this file to see how far it gets doing the backup, if it does anything at all. ■

Where can I turn?

Linux has gained a reputation for being obtuse and its support characterised as consisting of little more that a patronising, "Read the manual!" Well, this just isn't true. Andy Channelle shows us where to look for great advice...

A computer operating system is a complex beast and no one can be expected to know everything that's going on under the bonnet, so eventually you're probably going to need a little hand-holding. But where's the best place for you to get it?

If you've bought a boxed distribution from the likes of SuSE, Red Hat or Mandrake, your first port of call should be the manual. These are often graded to provide easy access to, for instance, a Quick Start guide, a more detailed system manual and, in the case of SuSE, a book covering a good selection of the apps that feature in the distro. The real problem with the printed word is that it 'captures a moment' that is soon passed and applications, desktops and OSs can change rapidly. This is where e-docs come in.

It has become common, in these cost-conscious times to forego the expense of a printed manual in favour of electronic documentation and if you've downloaded an ISO or application from the net, the chance of getting hard copy documentation is incredibly small, unless you print it yourself from the readme or help files that often accompany an app. Readmes tend to cover late-breaking issues, known bugs and so on, and are an important resource to investigate before submitting a report on a bug you've discovered. The other options are to hit the Help drop-down that usually appears on the right side of a menu bar (if you're using an app with a GUI) or combing through man pages.

The Help files are self-explanatory, but the **man** command goes back to the origins of Unix and is used – via the command line or something like *xman* which adds a GUI – to find, view and print online manuals. For example, typing **man mount** at the command prompt will search for documents containing references to the 'mount' command. As with most search engines, you can limit your query to a particular section or keyword. Type **man man** for a list of all options for the command, as well as instruction on how to hone your searches and avoid the scatter-gun approach. Man pages are not written to win literary awards – they get straight to the point and are best suited to uncovering the various options/syntax of a command. They can also be used for pairing up a function with its associated command – just search using the function as a keyword.

If you've exhausted the paper and online manuals, the next stop in the quest for satisfaction should be the website of your distribution vendor. Boxed products tend to include some form of telephone or online support, with the most intensive options covering the installation and bedding down period (the first 60–90 days). Following this period, all of the big vendors offer a combination of free and paid support, varying from general configuration queries to highly tailored support designed for your particular network, hardware and software setup. The latter is obviously designed with enterprise and small business users in mind, but if you're really stuck, a small 'per-incident' charge may pay dividends if it gets your system off the floor.

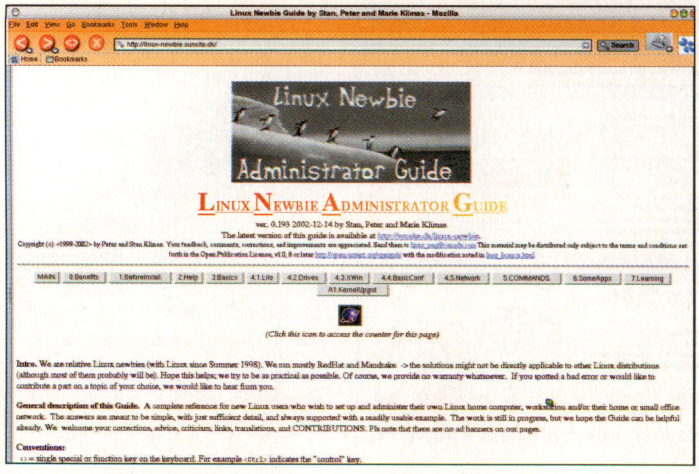
On your own?

Most vendors will also have some form of forum support system where other users and company experts may offer solutions to your problems. Though forum-based support is usually free and open to anyone, it does have its downsides: your answer may be a long time coming, for instance, and depend on a fellow user having encountered the same problem that you have.

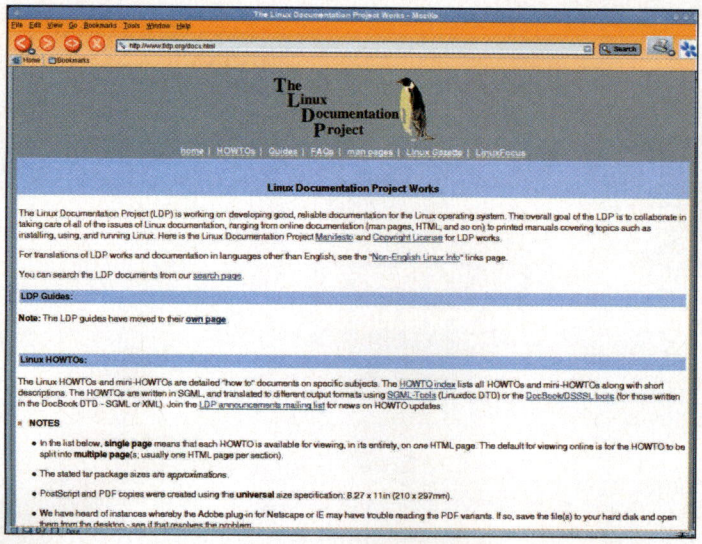
Any potential solutions may also be figured out on hardware and software that's far different from what you're currently using. In many cases you'll get a timely response, but if your job, company or thesis depends on a rapid answer, you may find that paying for support is your best option.

For some users, solving a problem is part of the enjoyment (within reason) of using a computer. If this is you, browse the Linux Documentation Project at **www.tldp.org/docs.html** for a real eye-opener. It contains a fully-featured set of man pages (with links to a couple of decent search engines) and a vast store of guides, Frequently Asked Questions and HOWTOs. The guides alone are a worthy addition to your support armoury, with details on everything from Advanced Bash Scripting to Kernel Module Programming.

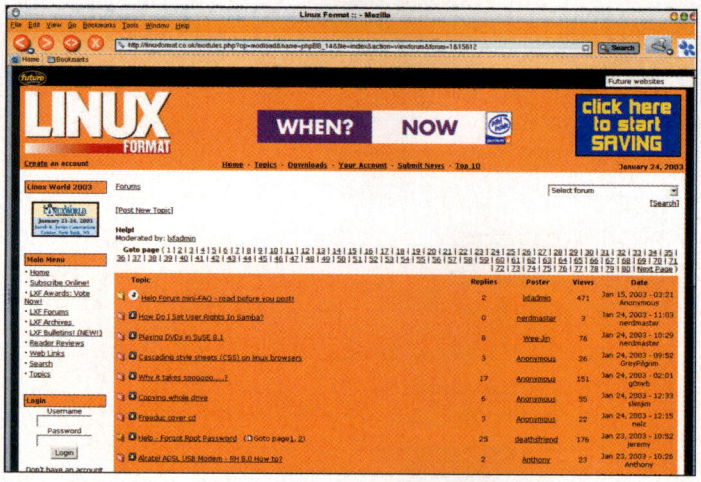
Some of these guides are written with new users in mind, so don't launch into reams of coding without


Linux Newbie Guide by Stan, Peter and Marie Klimas - Mozilla
Linux Newbie
Administrator Guide
LINUX NEWBIE ADMINISTRATOR GUIDE

The Linux Newbie Administrator Guide aims to take you from decision to installation and beyond.


The Linux Documentation Project
home | HOWTOs | Guides | FAQs | man pages | Linux Gazette | LinuxFocus
Linux Documentation Project Works
LDP Guides:
Linux HOWTOs:
NOTES

The Linux Documentation Project is a central repository for hundreds of helpful documents, for everyone from newbies to experienced sysadmin.



The *LXF* forums are manned by knowledgeable and friendly Linux users.

preamble. Of particular note is the Linux Newbie Administrator Guide: **http://linux-newbie.sunsite.dk/**. It starts with very simple questions covering why you may want to adopt Linux in the first place, moves through acquiring a distribution, then installation and boot issues, before covering the advantages of Open Source and proprietary software, networking and finally a user-friendly guide to kernel upgrading. If you're considering migrating to Linux, this is the next best thing to a very thick 'How To Do Everything In Linux'-style book. And, of course, it saves you lots of money.

In between the terse man pages and the verbose LDP guides lies the world of HOWTOs, which do exactly as their name suggests. HOWTOs and their mini counterparts (which are shorter tutorials on more specific issues) contain detailed guidance on aspects of configuring and using Linux. As an example of their range, the Printing HOWTO covers everything from general printing issues to the Print2Win mini HOWTO which details how to set up your Linux box to output to Windows print share.

HOWTOs vary in both style and substance. The USB Digital Camera HOWTO, to take a random example, provides details not just of how to get your camera up and running, but covers what is going on behind the scenes when you run the included script, so you can check that things are happening as they should, and offers a selection of troubleshooting tips if things go wrong.

HOWTOs are written in SGML but are available as plain text, Postscript, PDF or HTML files. They are usually released under the GNU Free Documentation Licence which allows redistribution and modification as long as the altered text is released under the same licence. However, it is becoming increasingly common to write HOWTOs using the Wiki system of community collaboration, where anyone can modify the text by simply hitting a link at the bottom of the page.

LXF Forum and LUGs

Sometimes it can seem there is nowhere to turn, especially if your problem appears to be too specific for the general guides or HOWTOs. You could write into the Answers page of our sister publication, *Linux Format*. As the magazine has a longish lead time and a lot of requests, your question may take a while to be answered, but the solutions are going to be thorough and well researched.

The alternative is to visit the magazine's website and forums at **www.linuxformat.co.uk**, which has a section dedicated to technical queries, hardware, programming languages and general help. As well as being able to call on lxfadmin (when there's no pressing deadline) and the ever-present 'anonymous', the forums are also frequented by the likes of Jeremy, Nelz, Fingers99, Rhakios, Erin and many others who are brimming with knowledge and experience of using Linux in a wide variety of commercial and home situations.

Of course, the Linux experience isn't limited to the *LXF* forums. Linux User Groups often have mailing lists to supplement their RL activities (check out the LUG pages). **www.lug.org.uk** **www.linuxforums.org** and **www.yourlinux.co.uk**, among many others, have extensive forums, while entities such as IBM, Mozilla, Sun and OpenOffice have support systems in place to answer questions and help you sort out your problem.

We hope to have shown you by now that the support mechanisms for Linux are many and varied, and if you still haven't solved your difficulties, it might be time to consider hiring a consultant to help you out. However, that's another topic altogether... ■

Do your own documentation

Writing and printing

Documentation is one of the less glamorous aspects of Open Source development and as such is often left to the last moment, if it gets done at all. However, it's the perfect job for users who'd like to contribute to a project without having to learn another language. The big projects are always looking for users (who, after all, experience the application on a day-to-day basis) who can string a sentence together to help compile online help, tutorials and HOWTOs.

The real benefit here is that once you've 'been through it', pulling together all possible resources into a coherent whole, you will have a unique perspective on an issue from which others could benefit. It would be a shame to waste it...

If you find that you're having to print a lot of bulky docs, get some good quality paper and print on both sides. The most straightforward solution for PDF files is to print odd-numbered pages first, then reload the printer with the odd pages and print the even-numbered pages on the other side.

To save tinkering with printer settings too much, Plain Text files can be most easily printed on both sides of sheets by pasting them into your choice of word processing app or text editor that allows page breaks. Remember to check the required orientation of the paper first by doing a single-sheet test to avoid your documentation coming out overprinted or upside down.

"If you've exhausted the paper and online manuals, the next stop should be the website of your distribution vendor."

User Groups

LUGs worldwide are full of members keen to help with your problems, discuss ideas and generally natter about all things Linux. You can find lots more information online at: www.lug.org.uk

1 HAMPSHIRE
URL www.hants.lug.org.uk
Contact Hugo Mills

2 BRISTOL & BATH
URL www.bristol.lug.org.uk

3 SCOTTISH
URL www.scottish.lug.org.uk

4 OXFORD
URL www.oxford.lug.org.uk
Contact Alasdair G Kergon

5 KENT
URL www.kent.lug.org.uk
Contact Kevin Groves

6 BRIGHTON
URL www.brighton.lug.org.uk
Contact Johnathan Swan

7 WORCESTERSHIRE
URL www.worcs.lug.org.uk

8 NORTHANTS
URL www.northants.lug.org.uk
Contact Kevin Taylor

9 ANGLIAN
URL www.anglian.lug.org.uk
Contact Martyn Drake

10 MILTON KEYNES
URL www.mk.lug.org.uk
Contact Denny De La Haye

11 DONCASTER
URL www.doncaster.lug.org.uk
Contact Andy Smith

12 MORAY
URL www.moray.lug.org.uk
Contact Stewart Watson

13 WEST WALES
URL www.westwales.lug.org.uk
Contact Dan Field

14 WOLVES
URL www.wolveslug.org.uk
Contact Jono Bacon

15 PETERBOROUGH
URL www.peterboro.lug.org.uk
Contact Steve Gallagher

16 EDINBURGH
URL www.edinburgh.lug.org.uk
Contact Alistair Murray

17 TYNESIDE
URL www.tyneside.lug.org.uk
Contact Brian Ronald

18 LEICESTER
URL www.leicester.lug.org.uk
Contact Clive Jones

19 GREATER LONDON
URL http://gllug.linux.co.uk/
Contact John Southern

20 SURREY
URL www.surrey.lug.org.uk
Contact Jay Bennie

21 CAMBRIDGE
URL www.cam-lug.org.uk

22 DEVON & CORNWALL
URL www.dclug.org.uk
Contact Simon Waters

23 FALKIRK
URL www.falkirk.lug.org.uk

24 MANCHESTER
URL www.manlug.mcc.ac.uk
Contact John Heaton, Owen Le Blanc

25 HERTFORDSHIRE
URL www.herts.lug.org.uk
Contact Nicolas Pike

26 WEST YORKSHIRE
URL www.wylug.lug.org.uk
Contact Jim Jackson

27 SHEFFIELD
URL www.sheflug.co.uk
Contact Richard Ibbotson

28 STAFFORDSHIRE
URL www.staffslug.org.uk

29 NORTH EAST
URL www.shofar.uklinux.net/NELUG

30 LONDON
URL www.lonix.org.uk

31 BERKSHIRE & THAMES VALLEY
URL www.sclug.org.uk

32 LIVERPOOL OPENSOURCE
URL http://linux.liv.ac.uk/_liv_linux_ug/
Contact Simon Hood

33 DEAL AMIGA CLUB
Email superhighwayman@hotmail.com
Contact John Worthington

34 CHESTERFIELD
Email spirelug@yahoo.co.uk
Contact Robin Needham

35 SOUTH DERBYSHIRE
URL www.sderby.lug.org.uk
Contact Dominic Knight

36 BELFAST (BLUG)
URL www.belfastlinux.cx
Email russell@belfastlinux.org

37 WILTSHIRE
URL www.wiltshire.lug.org.uk
Contact Jason Rudgard

38 SOUTH LONDON
URL www.sl.lug.org.uk
Email edo@perceptiondm.com

39 CHESHIRE
URL www.sc.lug.org.uk
Contact Anthony Prime – enquiry@sc.lug.org.uk

40 NORTH WALES
URL www.northwales.lug.org.uk
Contact Jonathan Cole

41 MIDLANDS
URL http://midlandslug.port5.com/
Contact Pete Thompson

42 CUMBRIA
URL www.cumbria.lug.org.uk
Contact Jamie Dainton

43 DORSET
URL www.dorset.lug.org.uk
Contact John Robinson

44 SHROPSHIRE
URL www.shropshire.lug.org.uk
Email shropshire@lug.org.uk

45 SOUTH WEST
URL www.southwest.lug.org.uk
Email southwest@lug.org.uk

46 SOUTH WALES
URL www.swlug.org.uk

47 NORTH LONDON
URL www.kemputing.net/lug/anlug-aims.html
Email jason@voyagercomputers.co.uk

48 MALVERN
URL www.malvern.lug.org.uk
Contact Greg Wright

49 HUDDERSFIELD
URL www.hud.lug.org.uk
Contact Dave Naylor – knocker@caramboo.com

50 NOTTINGHAM
URL www.nottingham.lug.org.uk

51 ST ALBANS & LUTON
URL www.lust.lug.org.uk
Contact Michael Culverhouse – mike@easily.co.uk

52 WREXHAM
Contact Paul Kersey-Smith
Email paul@pkls.fsnet.co.uk

53 PRESTON & LANCS
URL www.preston.lug.org.uk
Contact Phil Robinson

54 DERRY
URL www.derry.lug.org.uk

55 ISLE OF WIGHT
URL www.iow.lug.org.uk
Contact David Groom – info@iow.lug.org.uk

56 SCARBOROUGH
URL www.scarborough.lug.org.uk

57 BLACKBURN
Email matt@consultmatt.co.uk

58 YORK
URL www.york.lug.org.uk

59 LINCS
URL www.lincs.lug.org.uk

60 HULL
URL www.hull.lug.org.uk

61 WALTON-ON-THAMES
Contact William Mutch
Email rael@freeuk.com

62 GLOUCS & COTSWOLDS
URL www.gloucs.lug.org.uk

63 WEST OF SCOTLAND
URL www.wos.lug.org.uk

64 SOUTH STAFFORDSHIRE
URL www.staffs.lug.org.uk

65 MANSFIELD
URL www.mansfield.lug.org.uk

66 BORDERS
URL www.linux.bordernet.co.uk

67 BIRMINGHAM
URL www.sb.lug.org.uk

68 COVENTRY
Email info@coventry.lug.org.uk

69 NEWARK & LINCOLN
URL www.newlinc.lug.org.uk

70 BEDFORDSHIRE
URL www.beds.lug.org.uk

71 LINCOLN
URL www.lincoln.lug.org.uk

72 LOUGHBOROUGH
URL www.loughborough.lug.org.uk

73 EXETER UNIVERSITY
Email N.J.Murison@exeter.ac.uk

74 SUNDERLAND
Email thomas.croucher@sunderland.ac.uk

75 EAST YORKSHIRE
Email sharkonline@whatemail.com

76 CLEVELAND OPEN SOURCE GROUP
Email openlug@digitalmedia.co.uk

77 BEVERLEY
Email vladimir_lukyanov@hotmail.com

78 DUNDEE & TAYSIDE
URL www.dundee.lug.org.uk

79 SUSSEX
URL www.phpworld.co.uk/~swlug

80 WIGAN & ST HELENS
Email paulf.johnson@ukonline.co.uk

81 BRIXTON
URL www.communitytechnology.org.uk/~linuxhome

82 ST.ANDREWS, FIFE
URL www.standrews.lug.org.uk
Email stuart@nx14.com

83 NUNEATON
URL www.nuneaton.lug.org.uk

84 ISLE OF MAN
URL www.iom.lug.org.uk
Email helix@manx.net

85 AYLESBURY
URL www.aylesbury.lug.org.uk
Email drbond@educational-computing.co.uk

86 LANCASHIRE
URL www.lancasterlug.org.uk

87 EAST LONDON
URL www.eastlondon.lug.org.uk
Contact Jonathan Spriggs

NEW DETAILS:

88 ORMSKIRK
Email rob@northwestlinux.co.uk

89 HEREFORD
URL www.hereford.lug.org.uk/
Email rbjh@good-news.fsnet.co.uk

90 EAST HERTS
Email madtom1999@yahoo.com

91 SWINDON
Email nick.trueman@ntlword.com

92 MENAI
URL www.menai.lug.org.uk

93 ABERDEEN
Email mark@arricc.net

94 SHETLAND
URL www.shetland.lug.org.uk
Email c_s_s_butler@yahoo.com

NORTH STAFFORDSHIRE
Meeting on February 18 to kick-start a North Staffs LUG. Details on http://titanic.co.uk/staffslug

OTHER GROUPS:

B-LUX
URL www.b-lux.org/
Email blux-admin@lineone.net

FUENGIROLA GNU/LUG
URL www.fuengirola.lug.org.uk/
Email fuengirola@lug.org.uk

SCHOOLS
URL www.schools.lug.org.uk/

TR LINUX – TURKISH LINUX
URL www.trlinux.lug.org.uk/
Email c_s_s_butler@yahoo.com

YOUNG LINUX
URL www.young.lug.org.uk/

Well, are you having fun yet?

Cast your mind back to the beginning of this magazine. We made the bold statement that using Linux was easier than you might have otherwise expected. I hope you found this to be the case.

In some ways it depends on your expectations. Everything is not going to be exactly the same as MS Windows, for example, because Linux isn't Windows, in the same way that the Macintosh operating system isn't Windows. Unlike either of those systems, though, you can customise Linux almost infinitely, so it is possible to make it more like either of them – or, more importantly, something that works better for you. Take a look in the KDE control panel for just a slice of some of the things you can change.

Until fairly recently, Linux also suffered from poor support from hardware manufacturers. The people who made graphics cards, modems, scanners, printers and so on just weren't interested in helping to create the drivers for Linux. There have been a lot of changes here, with people like ATI, NVidia, HP, Lexmark and Matrox providing good support for their Linux hardware, and more and more companies are taking it more seriously.

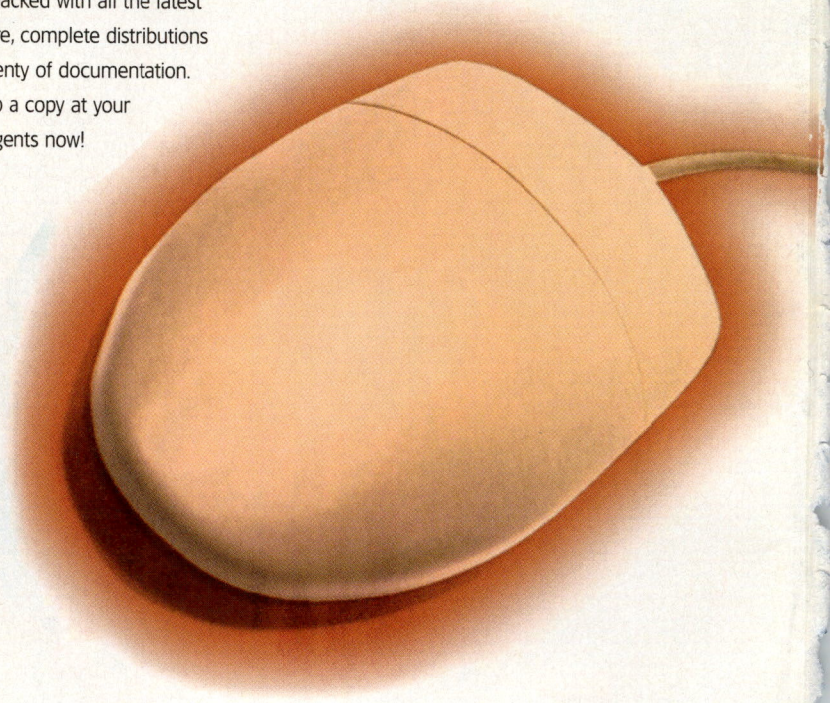
One of the great things about Linux is that it is constantly evolving, and for the most part you can keep your computer up to date with the latest software simply by downloading updates from the various distribution web and ftp sites.

Moving on

If you have tried Linux and decided that you want to keep using it, where should you go from here? This magazine has given you a basic grounding in some of the aspects of the operating system. Hopefully you'll have learnt how to use it sufficiently well to accomplish some good results with the free software included.

One thing you could consider is attending a local LUG meeting. We have covered the location of the major user groups in this magazine, and they are great places to learn new things, get help with your problems and meet other Linux users.

To keep up with what's happening in the Linux world, and for pages of advice, hints and more tutorials covering every aspect of Linux, we hope you'll be picking up *Linux Format* magazine on a regular basis. Every issue also comes with several CDs or a DVD, packed with all the latest software, complete distributions and plenty of documentation. Pick up a copy at your newsagents now!



Look for all this regularly in *Linux Format*...

The essential news and reviews...

Pages of tutorials covering all aspects of Linux...

The latest software on CDs or DVD!

When success becomes a burden, you know you're hosting with the wrong people.

Think of Linux hosting and you think of Rackspace. We pioneered Linux hosting and remain the world's largest Linux host. Why? You get access to the most passionate Linux support team anywhere. Free, unlimited access, 24 hours a day, 365 days of the year. No voicemail, no answering machines and 99.999% guaranteed network uptime. We call this Fanatical Support™. This is why we are different.

To get this kind of service, all you have to do is go to our website, or ring up and speak to a real person. Do that and a piece of hosting heaven can be yours in less than 24 hours.

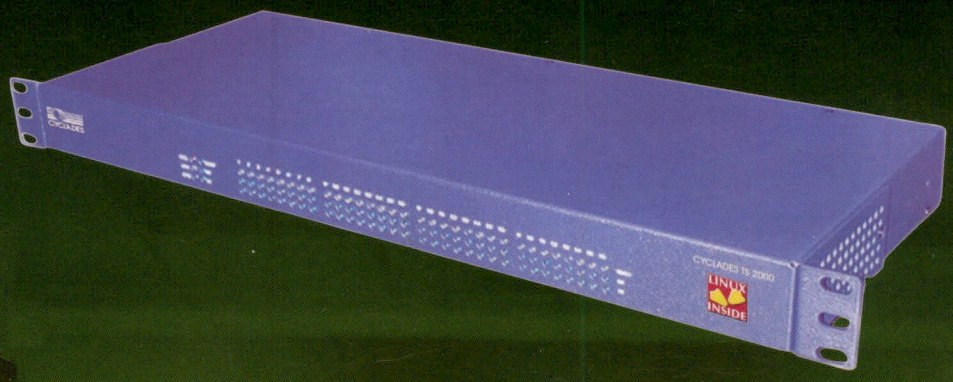
www.rackspace.co.uk

For a fanatical sales response, email alexander.harris@rackspace.co.uk or call 0800 085 3973 anytime.

100% human 100% of the time

rackspace
MANAGED HOSTING

Guess what Yahoo!® uses to manage their servers?

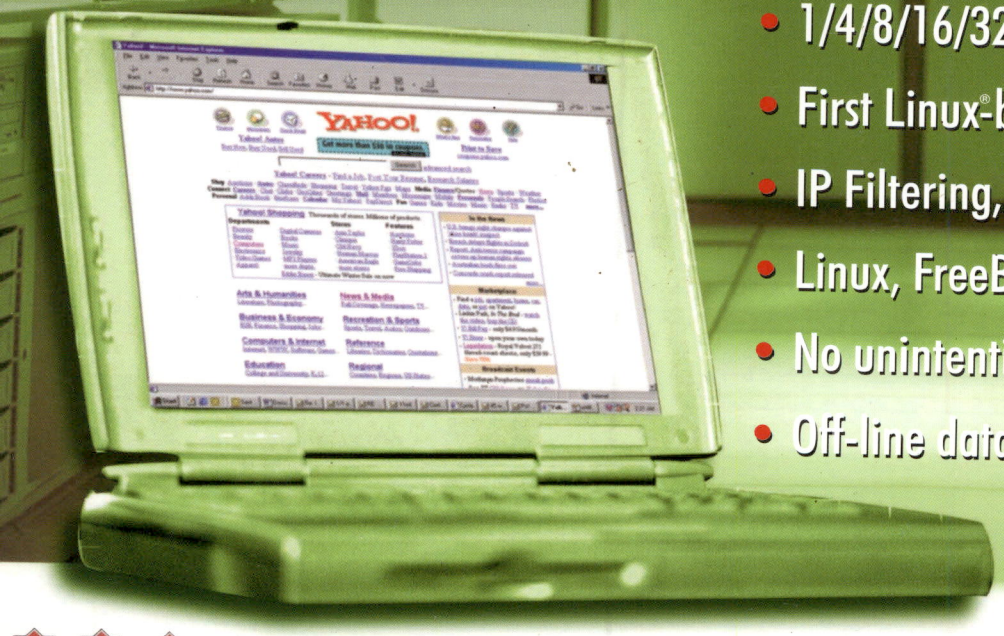


"The Cyclades-TS Series of Console Access Servers provides the highest port density and security at a very competitive price. By using Linux as the embedded OS, it offers the flexibility required to manage our dynamic environment. The Cyclades-TS is a key element to help us keep our servers up and running." - ***Pete Kumler,*** *Manager of Site Operations, Yahoo! Inc.*

Cyclades-TS Series

Console Access Server

- 1/4/8/16/32/48 RS-232 ports on 1U of rack space
- First Linux®-based Terminal Server on the market
- IP Filtering, RADIUS, and Secure Shell (SSHv2)
- Linux, FreeBSD, Sun, HP, and IBM compatible
- No unintentional breaks (Sun)
- Off-line data buffering



SOLARIS READY

LINUX INSIDE

Request for your FREE CAS booklet at www.cyclades.com

1999 READERS' CHOICE
2000 READERS' CHOICE
2001 READERS' CHOICE
1996 READERS' CHOICE
1997 READERS' CHOICE
1998 READERS' CHOICE

HE LEADER IN
LINUX
ONNECTIVITY

www.uk.cyclades.com

Voice : +44 (0) 1724 277 179

Fax: +44 (0) 1724 279 981

uksales@cyclades.com

CYCLADES®